The Dissertation Committee for Arvin Ebrahimkhanlou Certifies that this is the approved version of the following Dissertation:

Advanced Pattern Recognition Techniques for Wave-based Structural Health Monitoring of Metallic Panels

Committee:

Salvatore Salamone, Supervisor

Michael R. Haberman

Loukas Kallivokas

Michael Engelhardt

Trevor D. Hrynyk

Advanced Pattern Recognition Techniques for Wave-based Structural Health Monitoring of Metallic Panels

by

Arvin Ebrahimkhanlou

Dissertation

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Doctor of Philosophy

The University of Texas at Austin

December 2018

Dedication

To my wonderful family

Acknowledgements

I would like to express my sincere gratitude to my advisor Dr. Salvatore Salamone for his guidance and unwavering support. Also, special thanks are to Drs. Michael Engelhardt, Loukas Kallivokas, Michael R. Haberman, and Trevor D. Hrynyk for serving on my dissertation committee and providing me with their valuable advice. Throughout my Ph.D. studies, Dr. Salamone and Dr. Engelhardt have been central to my professional development and have dedicated a great deal of time preparing me to become a successful university professor in future.

This research would not have been possible without the financial support provided by the following institutions: the National Science Foundation (award numbers: CMMI-1333506 and CMMI-1551138), the Office of Naval Research (grant number: N00014-17-1-2367, Program Director Dr. Ignacio Perez), external fellowship form the American Society for Nondestructive Testing (ASNT), and several internal support through The University of Texas at Austin in the forms of startup funds, prestigious fellowships, and travel grants.

I would like to also thank my professors and colleagues in my previous academic institutions, including Dr. Sargur N. Srihari who taught me advanced machine learning and pattern recognition techniques, and Dr. Alireza Farhidzadeh who motivated me pursue such techniques, as well as my professors in Sharif University of Technology who introduced me with research, teaching, and trained me as an engineer: Drs. Homayoon Estekanchi, Mohammad Ali Ghannad, Ali Bakhshi, Mohammad Mehdi Ahmadi, Vahid Khonsari, Amir Reza Khoei, and Ali Akbar Golafshani (deceased).

Gratitude is also extended to influential professors in my field of research who have contributed to my professional development and supported this research with their constructive comments and discussions at conferences, including Drs. Tribikram Kundu, Francesco Lanza Di Scalea, Didem Ozevin, Mohammad Jahanshahi, Piervincenzo Rizzo, Sourav Banerjee, and Kanji Ono. In addition, I would like to thank my mentors that advised me through ASNT mentoring program: Mr. Robert Shannon, Dr. Tsuchin Philip Chu, and Dr. Valery Godinez. Special thanks are also due to Dr. Shant Kenderian who established this mentoring program. It is also important to thank Dr. Gary Georgeson who is in the process of donating a piece of Boing aircraft to The University of Texas at Austin that will facilitate the next stage of the present research.

Special thanks are also to Dr. Engelhardt, the director of the Ferguson Structural Engineering Laboratory at The University of Texas at Austin, Dr. Oguzhan Bayrak, and Dr. Hrynyk for allowing me to collaborate in their research projects and publishing journal papers that are beyond the scope of this dissertation.

I would like to recognize the friendship and support of my friends, past and current fellow students, and colleagues: Brennan Dubuc, Apostolos Athanasiou, Stylianos Livadiotis, Alireza Farhidzadeh, Ehsan Dehghan Niri, Jongkwon Choi, Marco Manconi, Alessandro Musumeci, Giuseppe Battaglia, Korkut Kaynardag, Mohammed Ali Morovat, Hossein Yousefpoursadatmahalleh, Amir Reza Ghiami Azad, Farid Khosravikia, Marina C Nakajima, Gabriel Polo Gonzalez, My Kim Vu, Deyuan Zhang, Xinran Li, Caitlyn N Kallus, and Marco Munoz.

Above all, I would like to thank my lovely wife for her encouragement and patience as well as my beloved family for their unconditional support.

Preface

Research topic

This research focuses on guided wave-based structural health monitoring (SHM) of metallic panels ubiquitous in civil, aerospace, and naval structures. The ultimate goal of this research is the safety and integrity of these aging structures that are being used beyond their design load and design life. Toward that end, this dissertation develops analytical and data-driven models that allow processing and interpreting the reverberation of guided ultrasonic waves. These reverberations are due to multiple reflections of the waves from structural and geometrical features of metallic panels, such as boundaries, stiffeners, and fasteners. Traditionally, such reverberations have been a major source of false positives in wave-based SHM resulting in falsely detected, non-existing defects that undermine the effectiveness of the SHM system. Instead, this research leverages the reverberations to reduce the number of transducers that the SHM system uses for transmitting and receiving guided ultrasonic waves. The considered SHM system works in two modes: 1) active ultrasonic imaging and 2) passive acoustic emission (AE). However, more weight is given to the latter. Advanced pattern recognition techniques are used to detect, localize, and characterize structural defects, including through-thickness holes and simulated fatigue cracks. In addition, this research uses probabilistic approaches to quantify the uncertainty of damage localization in the passive AE mode.

Organization

The format of this dissertation consists of five journal papers, collated for the purpose of this dissertation, prepared, and accepted by the author and his supervisor. Of these five papers, four has been published, and the remaining one has been reviewed internally and approved to be submitted for a journal publication. The following reviews the synopsis of each paper and explains how they are related to each other.

Paper 1

Paper 1 (Ebrahimkhanlou et al. 2016b) presents a data processing approach to reduce the number of permanently attached piezoelectric transducers for guided ultrasonic imaging of plate-like structures in the active mode. This paper leverages reflected waves to reduce the number of transducers and expand the monitoring area to the entire under investigation plate. To this end, an analytical model, named multipath ray tracking (MP) is developed to estimate the envelope of scattered waves from defects, such as corrosion dents and fatigue cracks. In particular, the model tracks wave propagation paths from a transducer to a scatter (i.e. defect) and then from there to another transducer. These paths that may include mode conversions and edge reflections are used to appropriately delay and attenuate several echoes of the initial excitation. These late arrival echoes are summed together to estimate the reverberation patterns of the scattered waves. Then, the correlation between the estimated and experimentally collected reverberation patterns is used to generate a tomographic image of defects. The approach is validated on an aluminum plate instrumented with three low-profile piezoelectric transducers, and results are compared with two traditional algorithms: Delay-and-Sum (DS) and Minimum Variance (MV). Damage is simulated by placing two magnets on the two sides of the plate and drilling through-thickness holes. It is

shown that including Lamb wave reverberations improves the localization accuracy while making the use of a fewer number of transducers possible.

Paper 2

Paper 2 (Ebrahimkhanlou and Salamone 2017c) extends the application of the analytical model introduced in paper 1 (i.e. the MP model) to the passive SHM mode (i.e. AE). In addition to reverberation patterns, this paper leverages the multimodal and dispersive characteristics of guided ultrasonic waves to localize AE sources. This approach is implemented in three steps, in the first steps. First, the difference in the propagation velocity of two guided ultrasonic modes (first symmetric and first antisymmetric modes) is used to estimate source-to-sensor distances. Then, the MP ray-tracking model is used to reconstruct the reverberation patterns of AE waveforms. To perform this task, the model uses the estimated source-to-sensor distances as well as first arrivals in the recorded waveforms. Finally, the correlation between the reconstructed and recorded waveforms are used as a criterion to localize AE sources. Unlike traditional AE source localization algorithms, such as triangulation methods that require at least three sensors and are only accurate within the convex area covered by the sensors, this paper uses only one sensor and has no blind zone. To validate the approach, experiments are carried out on an aluminum plate similar to the one used in paper 1. In particular, standard pencil lead-break tests (a.k.a. Hsu Nielsen source) were used to simulate progressive damage, such as fatigue cracks. The results are promising, and the reported statistics prove the benefit of using reverberation patterns, multimodal characteristics, and dispersive behavior of AE waveforms.

Paper 3

Paper 3 (Ebrahimkhanlou and Salamone 2017e) probabilistically quantifies localization uncertainties in paper 2. First, uncertainty in the source-to-sensor distance estimation is quantified based on the Hinesburg uncertainties of the wavelet transform. Specifically, a total least squares (TLS) problem is formulated and an unscented transform (UT) is used to propagate the Hinesburg uncertainties though the TLS problem. After solving the TLS problem, the paper uses the Cramer-Rao bound of the solution to quantify uncertainties in the source-to-sensor distance estimation. To translate the uncertainties in terms of AE source location, in the next step, Monte Carlo simulations are used. These simulations propagate uncertainties over the MP ray-tracking model and the correlation test performed in paper 2. Finally, the paper uses a kernel density estimation (KDE) technique to calculate a confidence contour for the location of AE sources. To validate the approach, similar experiments to those carried out in paper 2 are used and promising results are achieved.

Paper 4

Paper 4 (Ebrahimkhanlou and Salamone 2018b) takes an alternative, data-driven approach to the analytical approach of paper 2. The objective is to extend the application of single-sensor AE source localization to metallic panels with rivet connections and stiffeners. In particular, two deep learning architectures are implemented and compared: 1) stacked autoencoders and 2) convolutional neural networks. Deep learning implicitly uses the information pertaining to the edge reflections of Lamb waves as well as their dispersive and multimodal characteristics. Specifically, the deep learning model directly learns such reverberation patterns, dispersive characteristics, and multimodal behavior from the continuous wavelet transform of AE waveforms. In this paper, the output of the

deep networks is the zone in which an AE source occurs. Such zones include areas around rivet connections, areas on the surface of metallic panels, and areas along the edges of the panels. To evaluate the performance of this approach, a stiffener was riveted to a similar plate used to validate paper 2. The results show that, similar to the analytical approach taken in paper 2, the deep learning approach has also no blind zone and can localize AE sources anywhere on the panels. However, unlike the analytical approach, the deep learning approach is no longer limited to simple plates and can localize AE source in metallic panels with geometric features, such as rivets and stiffeners.

Paper 5

Paper 5 (Ebrahimkhanlou and Salamone TBD) extends the deep learning approach developed in paper 4. In particular, a two-step framework is developed that both localizes and characterizes AE sources. The first step of this framework is similar to the one presented in paper 4. This step finds the zone in which an AE source has occurred (i.e. zonal source localization). However, the second step is novel. In this step, depending on the zone to which the source belongs, a deep network either finds the coordinates of the source or categorizes the source based on its source-to-rivet distance. In addition, the paper assesses the sensitivity of this deep learning-based framework to 1) the number of the sensors, and 2) the metallic panel used to train the deep networks. Specifically, the accuracy of coordinate-based localization is compared for one to four sensors, and three typical scenarios are considered in which the training and testing conditions of the deep networks are not identical: 1) the deep networks are trained on one metallic panel and tested on another, 2) the deep networks are trained on a portion of a metallic panel and tested on another portion of the same panel, and 3) the deep

networks are trained with one sensor and tested with another sensor on a slightly different location on the same panel. To evaluate the performance of the deep learning-based framework, experiments are carried out on two identical panels similar to the one used in paper 4. The results show that the two-step framework can localize and characterize AE sources even if it is tested in a different condition that it was trained.

Author's contribution

The author of this dissertation was the lead investigator conduction the research. The author was responsible of developing theories, designing and conducting experiments, analyzing the results, and writing the journal papers. The coauthors contributed by providing editorial comments.

Abstract

Advanced Pattern Recognition Techniques for Wave-based Structural Health Monitoring of Metallic Panels

Arvin Ebrahimkhanlou, Ph.D.

The University of Texas at Austin, 2018

Supervisor: Salvatore Salamone

Increasing loads on aging and deteriorating aerospace and naval structures, such as airplanes and marine vessels, their usage beyond the designed life, and the desire to reduce the downtime associated with their regular maintenance operations have all motivated research on structural health monitoring (SHM) methods. Among SHM methods, those based guided ultrasonic waves, which are excited and received by low-profile piezoelectric transducers, are one of the most promising candidates for detecting, localizing, and characterizing structural defects. Despite the significant development of these SHM systems, very few, if any, have been implemented in real structures. One major reason for this limited implementation is due the difficulty of the processing and interpreting the reverberation patterns of guided ultrasonic waves. Such reverberations are due to multiple reflections of the waves from structural and geometrical features, such as boundaries, stiffeners, and fasteners. Therefore, the primary goal of this research is to overcome this challenge by advancing pattern recognition techniques and analyzing the patterns of edge-reflected guided-ultrasonic reverberations in thin metallic panels.

The objective is to leverage such patterns to improve the accuracy of current damage localization algorithms and reduce the number of required sensors. Specifically, two damage localization modes are considered: active ultrasonic imaging and passive acoustic emission. However, this dissertation gives more weight to the latter. For both active and passive modes, an analytical model are developed to simulate the patterns of edge-reflected, guide-ultrasonic reverberations. For the passive mode, a probabilistic framework is also developed to quantify the systematic uncertainties associated with this reflection-based localization approach. In addition, deep learning based, data-driven approaches are used to extend the application of the passive mode to metallic panels with rivet-connected stiffeners and allow characterizing the defects. For validation, experiments are conducted on rectangular aluminum panels with square-cut edges. The results show the effectiveness of the developed pattern recognition approaches in detecting, localizing, and characterizing structural defects, such as simulated fatigue cracks, with significantly fewer number of sensors. The knowledge gained in this investigation contributes to the condition awareness of metallic panels.

Table of Contents

List of Tables

List of Figures

PAPER 1

Damage Localization in Metallic Plate Structures using Edge-Reflected Lamb Waves

Arvin Ebrahimkhanlou, Brennan Dubuc, and Salvatore Salamone[1]

1.1. SYNOPSIS

This paper presents a model-based guided ultrasonic waves (GUW) imaging algorithm, in which multiple ultrasonic echoes caused by reflections from the plate's boundaries are leveraged to enhance imaging performance. An analytical model is proposed to estimate envelope of scattered waves. Correlation between the estimated and experimental data is used to generate images. The proposed method is validated through experimental tests on an aluminum plate instrumented with three low profile piezoelectric transducers. Different damage conditions were simulated including through-thickness holes. Results are compared with two other imaging localization methods, that is, Delay and Sum (DS) and Minimum Variance (MV).

Keywords: Guided ultrasonic waves, ultrasonic imaging, tomography, multipath, edge reflection, pattern recognition, plate structures, structural health monitoring

---

[1]This chapter have been previously published as: Ebrahimkhanlou, Arvin, Dubuc, Brennan, and Salamone, Salvatore (2016). "Damage localization in metallic plate structures using edge-reflected lamb waves." Smart Materials and Structures, 25(8), 085035. DOI: https://doi.org/10.1088/0964-1726/25/8/085035
According to the publisher's website: https://publishingsupport.iopscience.iop.org/questions/quick-check-guide-current-author-rights-policy/ accessed on July 25, 2018 inclusion of accepted manuscripts in a research dissertation is permitted. In addition, the dissertation advisor has allowed the direct reproduction of this paper. This is a non-typeset version of the accepted manuscript except for the numbering of sections, figures, tables and equations that have been adapted from the original to satisfy the formatting requirements for a dissertation. The author of this dissertation was the lead investigator conduction the research. The author was responsible of developing theories, designing and conducting experiments, analyzing the results, and writing the journal paper. The coauthors contributed by providing editorial comments.

1.2. INTRODUCTION

Plate-like structures are ubiquitous in the aerospace, marine, and civil industries. Examples include, wings and fuselage in aircrafts, ship hulls, etc. Corrosion, fatigue cracking, and impacts are some of the most common threats to these structures. Structural health monitoring (SHM) by guided ultrasonic waves (GUWs) has emerged as a potential high impact technology for reducing life-cycle costs and improving the safety of these complex structures (Clarke et al. 2009; Rose 2014; Wang et al. 2004; Wilcox 2003; Yu and Giurgiutiu 2008). In the last few years several efforts have been made to combine GUW signals, collected by a sparse array of piezoelectric transducers, with imaging algorithms for damage localization in plate-like structures. Commonly, these algorithms use differential signals (i.e. subtraction of baseline data from subsequently measured signals) to isolate effects of damage on GUW measurements. Several techniques including optimal baseline selection (OBS) and baseline signal stretch (BSS) are developed to compensate for the effect of changing environmental (e.g. temperature) and working conditions (e.g. load) on GUW measurements (Croxford et al. 2010; Dubuc et al. 2016; Mazzeranghi and Vangi 1999; Salamone et al. 2009; Shi et al. 2013). These techniques provide a base that can be combined with the most damage detection and localization algorithms. A number of strategies have been developed to process the differential signals and improve imaging performance, including: (i) delay-and-sum (DS) approaches (Bagheri et al. 2014; Fendzi et al. 2016; Hall and Michaels 2010; Michaels et al. 2008; Sharif-Khodaei and Aliabadi 2014; Wang et al. 2004) whereby differential signals are delayed according to their time-of-flight (TOF) to (or from) each pixel; (ii) model-based approaches, in which a propagation model is used to generate a data set of scattered signals; then simulated scattered signals are compared with differential signals to generate the image (Kim and

Chattopadhyay 2015; Levine and Michaels 2013; Wang et al. 2010), and (iii) statistical approaches based on data analysis techniques applied to probabilistic features of the signals' data points (Flynn et al. 2011; Moustafa and Salamone 2012; Rojas et al. 2015).

Most of these algorithms use only the first arrival of the scattered waves, ignoring late arrival echoes and reverberations. However, such late arrivals might contain useful information that can be leveraged to improve imaging performance while using fewer transducers than traditional approaches. Flynn et al. (2011) implicitly used late arrivals to train a stochastic model and showed that poor performance of imaging algorithms may be due to neglected late arrivals. Late arrivals were also implicitly included in recent data-driven based imaging algorithms (Tibaduiza-Burgos and Torres-Arredondo 2015), and for damage localization in cylindrical structures such as pipelines (Dehghan-Niri and Salamone 2015; Leonard and Hinders 2003). Harley and Moura (2013) used ray tracking and included reflections in their optimal baseline subtraction method. They called late arrivals multipath reflections. Recently, Hall and Michaels (2015) used wave field data collected by a laser Doppler vibrometer (LDV) to incorporate multipath reflections into an imaging algorithm. Their approach showed significant improvement in image reconstruction even when using a smaller number of sensors than traditional elliptical imaging algorithms. However, the required LDV scan for baseline collection limits the number of times the baseline is collected during the service life of the structure, and it may increase the overall cost of the monitoring system.

In this study, a novel model-based imaging algorithm is presented which incorporates late arrival reflections from boundaries. Inclusion of edge reflections into the model can provide high performance imaging while using fewer transducers than

conventional techniques. Preliminary versions of this study by the authors are available (Ebrahimkhanlou et al. 2015, 2016a).

The paper is organized as follows. First, assumptions and theoretical aspects of the proposed multipath model are presented. Then, the correlation imaging algorithm is explained, followed by experimental results. Finally, concluding remarks and future studies are discussed.

1.3. MULTIPATH MODEL

The multipath (MP) model, described in this section, estimates patterns of multiple echoes in scattered guided ultrasonic waves (GUW). The model calculates the propagation path of each echo to estimate its time of arrival and attenuation. These paths generally consist of: (1) a first segment between actuator and scatterer (*segment I*), and (2) a second segment between scatterer and receiver (*segment II*). On each of these two segments, multiple edge reflections may take place, but mode conversion may only happen at the scatterer. The MP model is implemented in three major steps: (*i*) ray tracking, (*ii*) mode tracking, and (*iii*) signal envelope estimation. First, a ray tracking algorithm is used to calculate the multiple paths taken by GUWs to travel from an actuator to a scatterer, and from the scatterer to a receiver. Second, multiple combinations of wave modes traveling on the same path are tracked. Then, travel distance, number of reflections on each path, and modes combinations are used to delay and attenuate the echoes accordingly. Finally, all echoes coming from different paths and different mode combinations are superimposed to estimate the envelope of the scattered waves. The inputs to the MP model are the plate's geometry and properties, the transducer coordinates, the excitation frequency, the baseline

signals, and the location of a simulated scatterer. The necessary steps to implement the proposed model are outlined in the following subsections.

In this study only a single point-like defect that results in guided wave scattering is modeled. Continues monitoring is assumed to ensure early damage detection. A new baseline is assumed to be collected after observation of each emerging defect. Therefore, only the most recent defect among the multiple existing defects is modeled. Multiple scattering (scattering of scattered waves at another damaged locations or at small fastener holes) is neglected, because amplitude of the secondary scattered waves is an order of magnitude smaller than the amplitude of the primary scattered waves.

1.3.1. Ray Tracking

Ray tracking is an algorithm used to calculate wave propagation paths connecting an acoustic source (i.e., actuator, or scatterer) and a receiver. These paths can be broadly divided into one direct path and multiple indirect paths. In this paper, the number of reflections occurring on each path is defined as the *reflection order*. Therefore, the direct path is assumed to have *zero-order* (Figure 1.1(a)). Figure 1.1(b-c) show an example of an indirect path having a first order and second order, respectively. In developing the ray tracking algorithm, certain assumptions will need to be made that limit the scope of systems that the algorithm can handle. These assumptions are explained below.

Linear elastic material behavior is considered. The excitation frequency is selected below the first cutoff frequency, so just the two fundamental modes, symmetric (S_0), and anti-symmetric (A_0) can propagate. The only propagating shear horizontal mode in this range of frequency is the SH_0 mode. Because the transducers are insensitive to this mode, its amplitude is negligible compared to S_0, and A_0. The two modes are reflected by the

plate edges. However, the incident S_0 mode is reflected as S_0 only, and the A_0 mode is reflected as A_0 (Gunawan and Hirose 2007; Torvik 1967). The propagation direction of the reflected waves is governed by Snell's law (Gunawan and Hirose 2007):

$$k_i \sin(\theta_i) = k_r \sin(\theta_r) \tag{1.1}$$

where k_i and θ_i are the wavenumber and angle of the incidence wave, respectively, while k_r and θ_r are the wavenumber and angle of the reflected wave (see Figure 1.1(c)). Since no mode conversion occurs, the wavenumber remains the same. Therefore, the incident and reflected angles are also the same. In other words, plate edges act as mirrors so the path taken from a source to an edge and then to a receiver can be traced by connecting the receiver to the mirrored location of the source with respect to the boundary. Once the line reaches the boundary, the rest of the path can be traced by connecting the intersection point to the actual location of the source, as shown in Figure 1.1(b-c). The flowchart in Figure 1.2 summarizes the overall procedure for finding all possible paths connecting an acoustic source and receiver. Also, detailed steps are presented in the form of a computer pseudo code in section 1.10. The algorithm gets the geometry of the plate and the location of the source and receiver. Then, it returns a dictionary of all possible paths up to the specified maximum reflection order on each path.

Table 1.1 summarizes a typical output of the algorithm for the example shown in Figure 1.1. In particular the following variables are presented for all possible paths up to the second order: number assigned to a path segment (q'); order of the path segment (o); sequence of boundaries that interact with the path; coordinates for first and second reflection location (r_1 and r_2 on Figure 1.1(c)); and travel distance of the path segment (d).

Finally, as an incident wave approaches the defect, it is scattered in all directions. In other words, the scatterer acts as a source. Therefore, the same procedure for tracking

the path connecting the exciting transducer to the scatterer can be used for tracking a path connecting the scatterer to a sensor. Similar to the first portion of the path, edge reflection may occur during the second portion. All zero, first, and second order paths from an actuator to a scatterer (i.e., segment I), and from the scatterer to a sensor (i.e., segment II) are shown in Figure 1.3.

In this paper, a parameter Q defines the total number of all possible paths between actuator-scatterer and scatterer-sensor. For the case shown in Figure 1.3, Q=13x13=169. The reflection order on segment I and II of these Q paths are stored in two vectors of size $Q\times 1$: $\mathbf{o}^{\mathrm{I}}, \mathbf{o}^{\mathrm{II}}$. These data are retrieved from the output of the ray tracking algorithm for each segment. Similarly, $\mathbf{d}^{\mathrm{I}}$ and $\mathbf{d}^{\mathrm{II}}$ are $Q\times 1$ vectors containing travel distance information. In this notation, superscripts I and II indicate segment I and II, respectively.

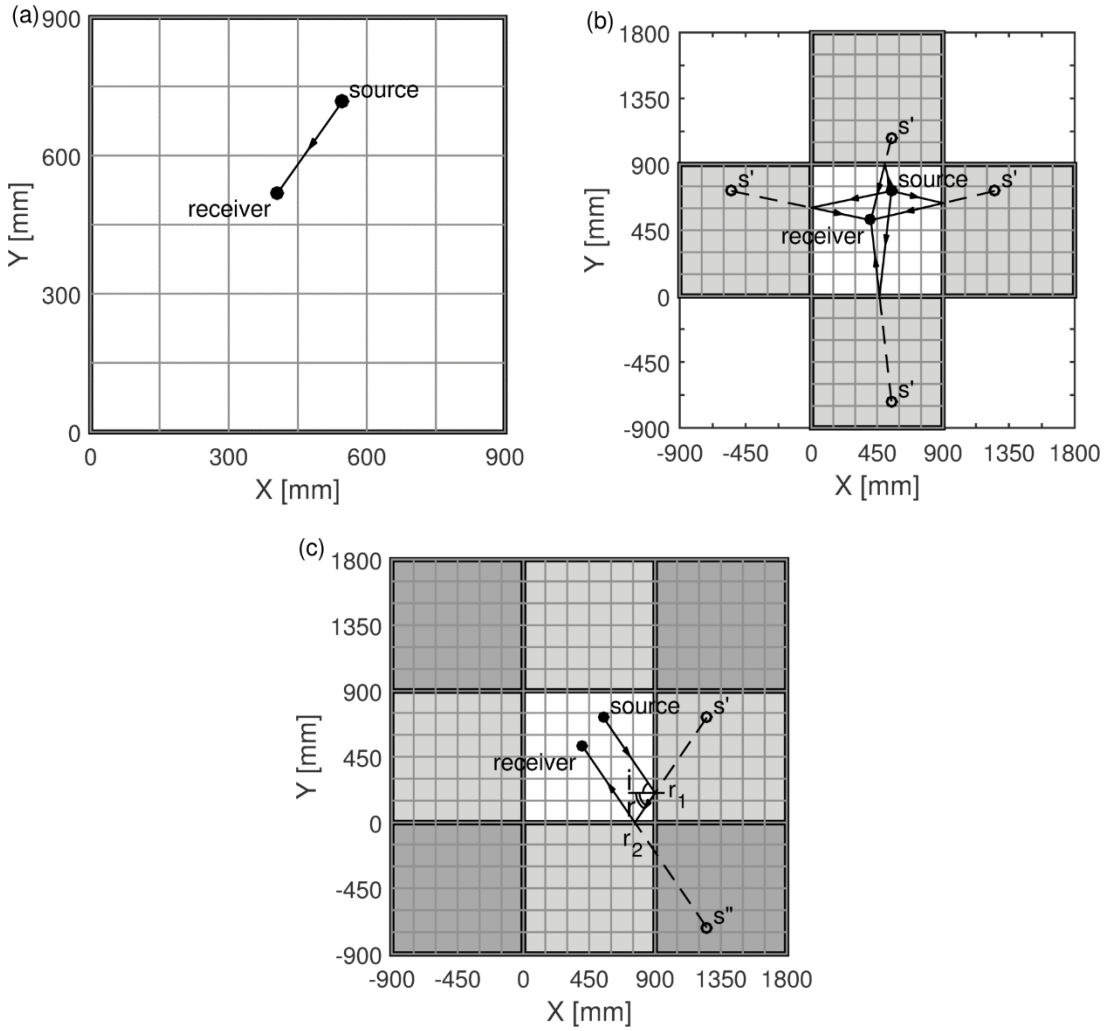

Figure 1.1: Ray tracking from a source to a receiver, virtual wings are shown in gray (a) a zero order path (b) first order paths (c) a second order path; first and second reflections are shown with r_1 and r_2 ; incidence and reflection angles are shown for r_1.

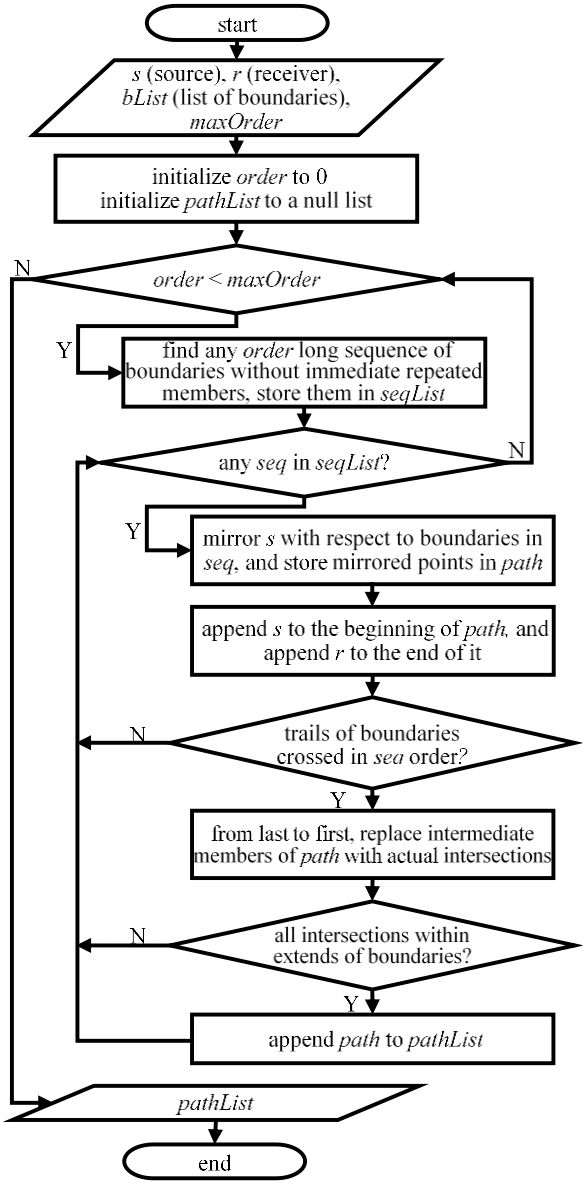

Figure 1.2: Flowchart of the ray tracking algorithm.

Table 1.1: Output of ray tracking algorithm for the source† and receiver‡ shown in Figure 1.1.

Path Number (q′)	Reflection Order (o)	Reflection Sequence	1st Ref. (r_1) (x,y) [mm]	2ed Ref. (r_2) (x,y) [mm]	Distance (d) [mm]
1	0	[]	-	-	248
2	1	[top]	(511,900)	-	562
3	1	[right]	(900,645)	-	856
4	1	[left]	(0,613)	-	990
5	2	[top,right]	(817 , 900)	(900,846)	994
6	2	[top,left]	(251,900)	(0,759)	1111
7	1	[bottom]	(473,0)	-	1265
8	2	[right,bot.]	(900,209)	(762,0)	1506
9	2	[left,bot.]	(0,8)	(6,0)	1586
10	2	[top,bot.]	(541,900)	(460,0)	1604
11	2	[right,left]	(900,687)	(0,577)	1669
12	2	[left,right]	(0,671)	(900,578)	1954
13	2	[bot.,top]	(504,0)	(439,900)	2007

†source coordinates [mm]: (556, 729) ‡receiver coordinates [mm]: (413,527)

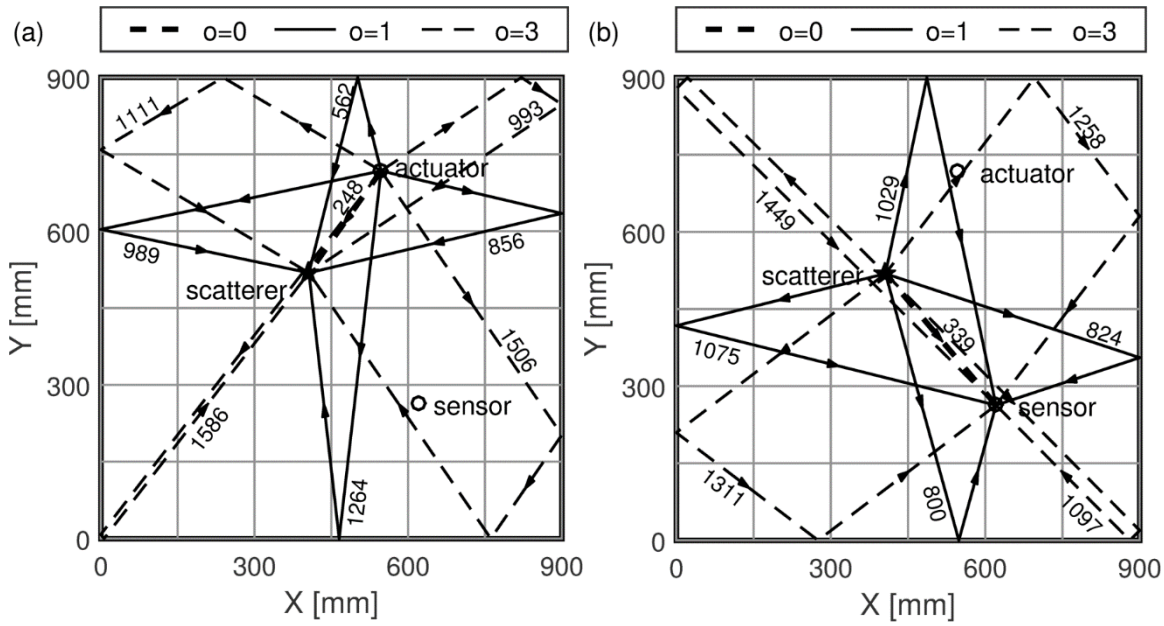

Figure 1.3: Direct, first and some of second order propagation paths with their travel distance, for the sake of the figure's clarity only the first nine paths are shown: (a) actuator to scatterer (segment I) (b) scatterer to sensor (segment II).

1.3.2. Mode Conversion Tracking

Mode conversion tracking keeps a record of mode conversions on each path. Below the first-cutoff frequency, where only the S_0 and A_0 modes can propagate on a path, mode conversions occur only during the scattering process, whereas no mode conversion is expected during edge reflections (see section 2.1). Two propagating modes on two segments of a path create four mode combinations which are listed in Table 1.2. In this table, m indicates the combination number, and m^{I}, m^{II} show the mode propagating on segment I and II of a path, respectively. The possible number of mode combinations on each path is indicated with M.

Figure 1.4 visualizes the propagation of the S_0 and A_0 modes on a path. The size of the plate and location of the actuator, scatterer, and sensor are the same as those in Figure 1.3. The wave fronts are shown at two different time instants, 40 and 100 μs, after the excitation. Figure 1.4(a) shows a time instant after the first reflection from the top boundary, but before the first scattering. Figure 1.4(b) visualizes multiple modes scattered in multiple directions from each incident wave at the scatterer.

Table 1.2: Mode combinations.

m	m^{I}	m^{II}
1	S_0	S_0
2	S_0	A_0
3	A_0	S_0
4	A_0	A_0

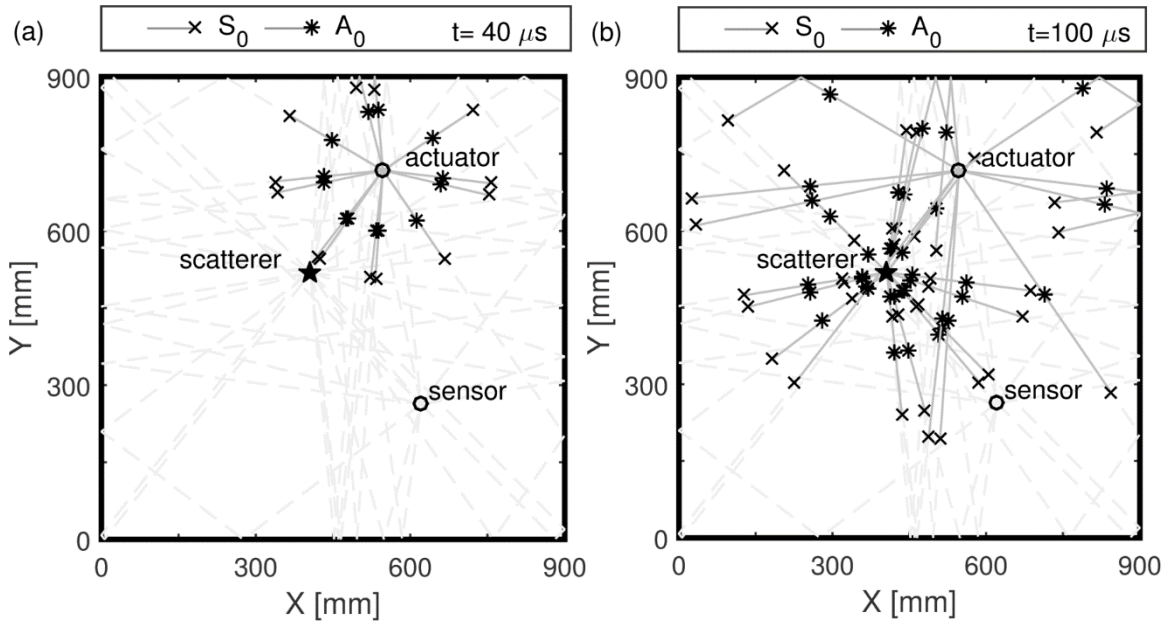

Figure 1.4: Propagation trace of S0 and A0 modes on a schematic plate, dash lines indicate up to second order paths that eventually end at the sensor. Group velocities of 5.329 mm/μs and 2.968 mm/μs are respectively used for S0 and A0 modes: (a) t=40 s, (b) t=100 μs.

1.3.3. Signal Envelope Estimation

Ray tracking and mode tracking provide all necessary information to estimate patterns in scattered waves. Given an actuator-sensor pair, p, and a simulated scatterer location (x, y), the envelope of the scattered wave $\hat{s}_{p,x,y}(t)$ can be estimated as the sum of each echo's envelope:

$$\hat{s}_{p,x,y}(t) = \sum_{q=1}^{Q} \sum_{m=1}^{M} \hat{s}_{q,m}(t) \tag{1.2}$$

where, $\hat{s}_{q,m}(t)$ is the envelope of the echo traveling on the path q and in mode combination of m. In Eq. (1.2), it is assumed that there is no overlapping between two echoes, for the sake of simplicity. For a toneburst excitation, this envelope can be assumed as an attenuated and time-shifted version of the excitation's envelope, that is:

$$\hat{s}_{q,m}(t) = \varphi_{q,m} \left| u_0(t - \tau_{q,m}) \right| \tag{1.3}$$

where $|u_0(t)|$ is the envelope of the excitation signal, $\tau_{q,m}$ is the time shift, and $\varphi_{q,m}$ is an element of an attenuation matrix $\mathbf{\Phi}$, defined as:

$$\mathbf{\Phi} = \mathbf{\Gamma} \circ (\boldsymbol{\alpha} \cdot \boldsymbol{\psi}^{\mathrm{T}}) \tag{1.4}$$

where $\mathbf{\Gamma}$ is a $Q \times M$ matrix associated to the attenuation due to reflections, $\boldsymbol{\alpha}$ is a $Q \times 1$ vector associated to attenuation due to spreading, and $\boldsymbol{\psi}$ is a $M \times 1$ vector associated to scattering. The operation ($\circ$) in Eq. (1.4) is a Hadamard product, and T is the vector transpose. A detailed description of attenuation parameters, and time-shift, is provided in the subsequent sections.

1.3.3.1. Spreading

A toneburst signal $u_0(t)$ at a distance d can be represented as (Levine and Michaels 2013):

$$u_d(t) = \mathcal{F}^{-1}\{(\frac{d}{d_0})^{-0.5}\mathcal{F}\{u_0(t)\}\mathrm{e}^{-\mathrm{i}(kd)}\} \tag{1.5}$$

where $\mathcal{F}\{\ \}$ is the Fourier transform; $\mathcal{F}^{-1}\{\ \}$ is the inverse Fourier transform; d_0 is a reference distance; i is $\sqrt{-1}$; and k is wavenumber (Levine and Michaels 2013).

By definition, group velocity, c_g at angular frequency ω_c is:

$$c_g(\omega_c) = \frac{\mathrm{d}\omega}{\mathrm{d}k}|_{\omega=\omega_c} \tag{1.6}$$

where ω is the angular frequency, and ω_c is the central angular frequency of the excitation signal. By substituting Eq. (1.6) in a first order Taylor series expansion of ω at ω_c:

$$\omega \approx \omega_c + (k - k_c)c_g(\omega_c) \tag{1.7}$$

and re-arranging Eq. (1.7):

$$k \approx \frac{\omega - \omega_c}{c_g(\omega_c)} + k_c \tag{1.8}$$

This approximation is valid for the plateaus of the dispersion curves where GUWs are less dispersive and can be approximated as non-dispersive. Substituting Eq. (1.8) into Eq. (1.5):

$$\begin{aligned} u_d(t) &= (\frac{d}{d_0})^{-0.5} \mathrm{e}^{-\mathrm{i}(k_c d - \omega_c \frac{d}{c_g(\omega_c)})} \mathcal{F}^{-1}\{\mathcal{F}\{u_0(t)\mathrm{e}^{-\mathrm{i}\omega \frac{d}{c_g(\omega_c)}}\}\} \\ &= (\frac{d}{d_0})^{-0.5} \mathrm{e}^{-\mathrm{i}(k_c d - \omega_c \frac{d}{c_g(\omega_c)})} u_0(t - \frac{d}{c_g(\omega_c)}) \end{aligned} \tag{1.9}$$

Based on the Euler equation ($\left|\mathrm{e}^{-\mathrm{i}\theta}\right| = 1$) and Eq. (1.9), the Eq. (1.5) takes the following form:

$$\begin{aligned} |u_d(t)| &= (\frac{d}{d_0})^{-0.5} \left| u_0(t - \frac{d}{c_g(\omega_c)}) \right| \\ &= A(d)^{-0.5} |u_0(t-\tau)| \end{aligned} \tag{1.10}$$

which consists of an attenuation term, $(d)^{-0.5}$ and a delay term τ. To further simplify the equation, $(d_0)^{0.5}$ can be considered as a multiplier A.

Because a scatterer acts as a source, the q^{th}-element of the vector $\mathbb{a}$ (attenuation due to spreading) in Eq. (1.4), consists of two geometric spreading terms, one from the actuator to the scatterer (segment I) and another from the scatterer to the sensor (segment II) (Hall and Michaels 2010). Therefore, by using two times the term $A(d)^{-0.5}$ in (1.10), attenuation due to spreading can be modeled as:

$$\begin{aligned} \alpha_q &= (\alpha_q^{\mathrm{I}})(\alpha_q^{\mathrm{II}}) \\ &= A(d_q^{\mathrm{I}})^{-0.5} A(d_q^{\mathrm{II}})^{-0.5} \\ &= A'(d_q^{\mathrm{I}} d_q^{\mathrm{II}})^{-0.5} \end{aligned} \tag{1.11}$$

where A' is a constant multiplier that is kept the same for all paths. Later, this constant is eliminated in the correlation imaging.

1.3.3.2. Edge Reflection

In this paper the Lamb wave attenuation due to reflection was modeled by reducing the amplitude of the incident wave by a coefficient γ, that is:

$$|u_r(t)| = (\gamma)|u_i(t)| \tag{1.12}$$

Different coefficients were used for the S_0 and A_0 modes. Specifically, $\gamma_{A_0} = 0.7$ was used for A_0, and $\gamma_{S_0} = 0.4$ for S_0. It should be noted that during the experimental tests, for safety reasons, sharp edges of the test plate were covered by duct tape which has been taken into account in selection of these coefficients. During the reflection process, a phase-shift between the incident and reflected waves occurs. Although this phase-shift can change the constructive or destructive interference of the two echoes, in this study, for the sake of simplicity, the phase shift was neglected.

Attenuation from consecutive reflections can be calculated by multiplying the amplitude loss coefficients of each of them. Therefore, the matrix $\mathbf{\Gamma}$ in Eq. (1.4) for attenuation due to reflection is constructed from multiple application of Eq. (1.12):

$$\Gamma_{q,m} = (\gamma_{m^{\mathrm{I}}})^{o_q^{\mathrm{I}}} (\gamma_{m^{\mathrm{II}}})^{o_q^{\mathrm{II}}} \tag{1.13}$$

where $\gamma_{m^{\mathrm{I}}}$ and $\gamma_{m^{\mathrm{II}}}$ respectively are amplitude loss coefficients for modes on segment I (m^{I}) and segment II (m^{II}); m^{I} and m^{II} can be retrieved from mode combination m using Table 1.2.

1.3.3.3. Scattering

Interaction of Lamb wave packets with a scatterer (e.g., crack, dent, hole, etc.) results in attenuation, phase shift, mode conversion, and scattering. However, as stated earlier, only mode conversion and scattering are considered in this study. The amplitude of different modes of scattered waves depends on many parameters including: the incident wave's mode, amplitude, frequency, and angle of incidence; damage type; and scattering

angle. In general, the scattered Lamb wave, $u_s(t)$ from incidence of $u_i(t)$ is (Levine and Michaels 2013):

$$u_s(t) = \mathcal{F}^{-1}\{\psi(\omega, m^{\mathrm{I}}, m^{\mathrm{II}}, \theta_i, \theta_s)\mathcal{F}\{u_i(t)\}\} \tag{1.14}$$

where θ_i and θ_s are respectively the incident and scattered angles; and $\psi(\omega, m^{\mathrm{I}}, m^{\mathrm{II}}, \theta_i, \theta_s)$ is far field scattering function of the damage. Although for some specific damage types and modes, theoretical or numerical far field scattering functions are derived (Grahn 2003), for the sake of simplicity in this study, the amplitude of scattered waves are considered independent of the incident or scattered angle. As discussed earlier, throughout this paper mode conversion to shear horizontal modes is considered as loss of energy, and reverse mode conversion back to Lamb waves were neglected. Therefore, without loss of generality, for angular frequencies close to the excitations' central angular frequency, ω_c, scattering functions for different mode combinations are assumed to be constant functions and their values are stored in vector: $\boldsymbol{\psi} = [0.01 \quad 0.02 \quad 0.02 \quad 0.1]^{\mathrm{T}}$. Based on the assumed constant valued far field scattering functions, the envelope of (1.14) is simplified to:

$$|u_s(t)| = \psi_m |u_i(t)| \tag{1.15}$$

It should be noted that the output of the model is used in correlation imaging, and in this application only the relative values of these constant functions affect the image, not their absolute values.

1.3.3.4. Time Shift

The time shift, $\tau_{q,m}$ in Eq. (1.3), depends on two things: the travel distance of a path; the group velocities of the modes traveling on the two segments of the path. Using Eq. (1.10) it can be concluded that:

$$\tau_{q,m} = \tau^{\mathrm{I}}_{q,m^{\mathrm{I}}} + \tau^{\mathrm{II}}_{q,m^{\mathrm{II}}} = \frac{\mathbf{d}^{\mathrm{I}}_q}{c_{g_{m^{\mathrm{I}}}}} + \frac{\mathbf{d}^{\mathrm{II}}_q}{c_{g_{m^{\mathrm{II}}}}} \tag{1.16}$$

where $\tau^{\mathrm{I}}_{q,m^{\mathrm{I}}}$ and $\tau^{\mathrm{II}}_{q,m^{\mathrm{II}}}$ are respectively travel time $c_{g_{m^{\mathrm{I}}}}$ on two segments of path q. Similarly, and $c_{g_{m^{\mathrm{II}}}}$ are the group velocity of each of the modes propagating on the first and second segments of the path, respectively.

1.4. CORRELATION IMAGING ALGORITHM

Damage localization in the MP imaging algorithm is achieved by means of correlation analysis between MP model estimations and experimental data. Eq. (1.2) estimates the envelope of the scattered echoes from a scatterer at (x, y). If the coordinates used in the model match with the coordinates of the actual damage, a high correlation between the estimated patterns and the envelope of experimentally recorded signals is anticipated. Therefore, correlation between MP model and experiments can be used as a measure for the existence of damage. In the correlation imaging algorithm, the MP model is run for the coordinates of each pixel. Then, the average of correlations between the estimation and envelope of recorded scattered waves from different transducer pairs is assigned to the pixel value:

$$\rho_{x,y} = \frac{1}{P}\sum_{p=1}^{P} \frac{\sum_{i=1}^{n_s}(s_p[i]-\bar{s}_p)(\hat{s}_{p,x,y}[i]-\bar{\hat{s}}_{p,x,y})}{\sqrt{\sum_{j=1}^{n_s}(s_p[j]-\bar{s}_p)^2 \sum_{k=1}^{n_s}(\hat{s}_{p,x,y}[k]-\bar{\hat{s}}_{p,x,y})^2}} \tag{1.17}$$

in this definition, s_p is the envelope of recorded scattered signals r_p from transducer pair p. The estimated scattered wave, $\hat{s}_{p,x,y}$ is digitized into n_s samples with the same sampling rate of s_p. Bars on top of parameters indicate average. Parameter P is the number of transducer pairs in the network. For a network of N transducers, $P = N(N-1)/2$.

In general, the localization accuracy of correlation imaging is more sensitive to the arrival time of each echo, rather than their relative amplitudes. Therefore, approximations in the model on these relative amplitudes does not impact the general trend of the image.

1.5. EXPERIMENTS

Experimental tests were carried out on a 900 mm x 900 mm x 3.18 mm aluminum plate to validate the proposed approach. More details about the plate are provided in Table 1.3. The plate was instrumented with three circular PZT transducers with a diameter of 12.7 mm. The size of the transducers was designed based on the thickness of the plate to ensure that their resonance frequency occurred in the plateau regions of the S_0 and A_0 dispersion curves. Signal generation and data acquisition were achieved with a National Instruments (NI), modular PXI unit. The unit includes an arbitrary waveform generator card (PXI 5412) and one multi-channel oscilloscope (PXI 5105). In addition, a high voltage linear amplifier with x10 gain was used to amplify the excitation signal, and two 40 dB preamplifiers (Olympus 5660B) were used to boost the received signals. The excitation signal was a 2.5-cycles toneburst at 0.18 MHz modulated by a Hanning window. A frequency sweep was carried out to identify the excitation frequency. Group velocities of the S_0 and A_0 modes were calculated from the dispersion curves shown in Figure 1.7(b). The figure shows that the selected frequency abides with the requirement of Eq. (1.7), and it is assumed non-dispersive.

In order to simulate damage conditions, circular magnets of grade N40 and diameter of 10 mm were placed on the plate, first outside (M_1) and then within (M_2) the convex area covered by the transducers. Baseline signals were recorded on the undamaged plate (i.e., without any magnets), and for each damage condition. Then a 1mm through-thickness

circular hole was introduced on the plate. Finally the diameter of the hole was increased to 2 mm. Locations of both the transducers and simulated damages are summarized in Table 1.4. An overview of the experimental setup is shown in Figure 1.5. Baseline subtractions were used to extract scattered waves. To minimize the systematic and random errors in the baseline subtraction phase a correction algorithm was developed. The algorithm uses the first arrival of the actual signal as a reference to correct sampling and amplitude mismatch errors. More details on the algorithm are presented in section 1.9.

The duration and relative amplitude of the S_0 and A_0 packets depends on the actuator, sensor and their bond with the structure. To compensate for the transducer induced effects, measurements on the baseline signal of each transducer pair is plugged in on the left-hand side of Eq. (1.10) and the input excitation is estimated from the right-hand side. Figure 1.6 shows the first arrival packets of the S_0 and A_0 modes. Because a Hanning-window was used to modulate the input excitations, a sinusoidal is fitted to the envelope of each packet. The amplitude, duration and number of cycles of each mode's packet are measured. Therefore, given the measured amplitude of each mode at the location of sensor, A_d, and propagation distance, d, which is the distance between two transducers, the initial amplitude of each mode's packet for each transducer pair can be calculated as:

$$A = A_d \sqrt{d} \tag{1.18}$$

The Hanning envelope for each mode's initial excitation can be reconstructed based on the measured number of cycles, n_c:

$$|u_0(t)| = A\sin(\frac{\omega_c}{2n_c}(t)) \tag{1.19}$$

Since in the correlation imaging only the relative value of the signals matters, value of A' can be assigned as an arbitrary multiple of A.

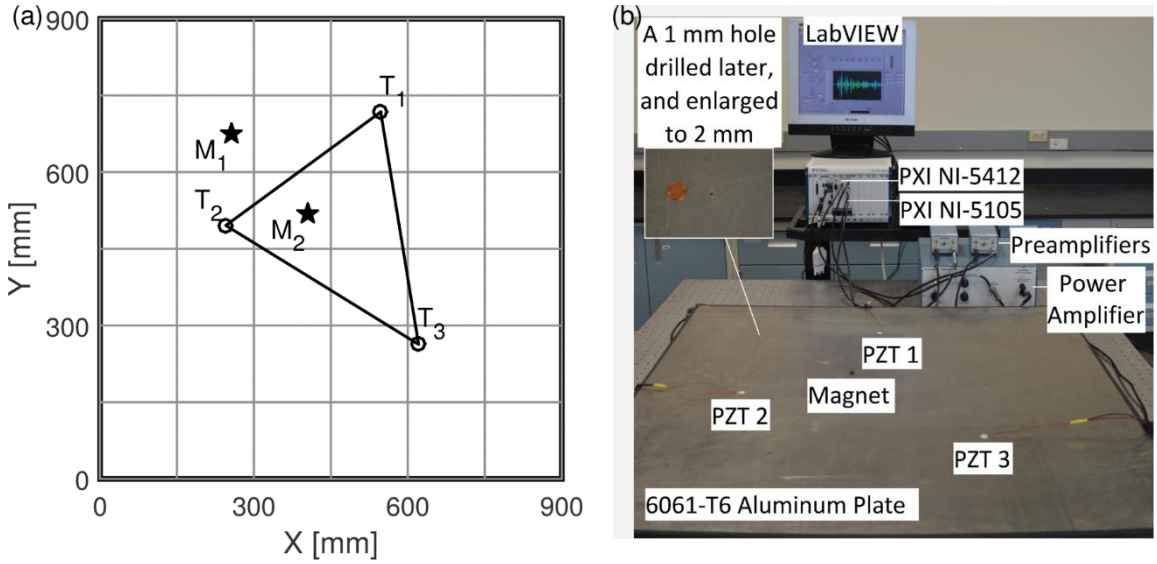

Figure 1.5: Experiments: (a) schematic overview, circles: PZTs, stars: simulated defects, the convex area covered by transducers is shown; (b) instrumentation set up.

Table 1.3: Material Properties.

Properties	Value
Material	Aluminum Alloy 6061-T6
Dimension [mm × mm × mm]	900 × 900 × 3.18
Elasticity Modules [GPa]	69
Poisson's ratio	0.33
Density [Kg/m^3]	2700

Table 1.4: Transducers and simulated damages locations.

Test	Point	X [mm]	Y [mm]	Type
all	T_1	556	730	Transducer
all	T_2	249	504	Transducer
all	T_3	631	267	Transducer
1	M_1	260	686	Magnet
2	M_2	413	527	Magnet
3	M_1	260	686	Through

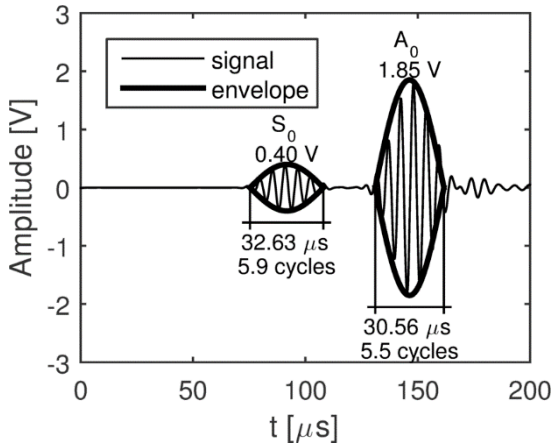

Figure 1.6: Baseline signal for T_1-T_2 pair; envelopes of the first arrivals of S_0 and A_0 are shown with their amplitude, duration and number of cycles.

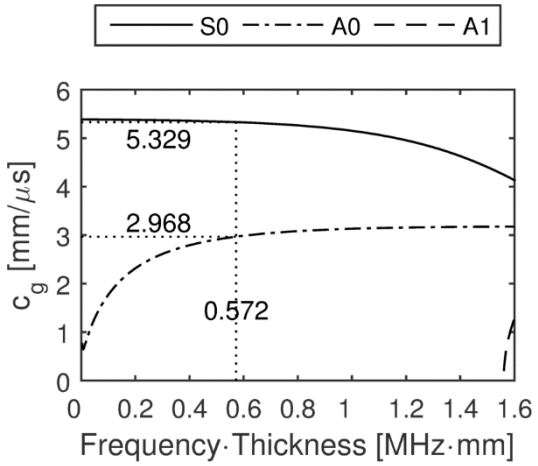

Figure 1.7: Dispersion curves for the plate's material, group velocities are shown for 0.18 MHz and thickness of the plate.

1.6. RESULTS

1.6.1. Envelope estimation

Figure 1.8 compares the envelope of scattered waves (differential signal) between the experimental data (s_p) and MP model calculation ($\hat{s}_{p,x,y}$). Three cases are given for damage condition M_1, with different actuator-sensor pair. The results show an overall good correlation between the experimental data and the calculated data. The mismatch between

the model and experiment is more pronounced at the peaks. A path associated with the dominant peak in Figure 1.8(a) which demonstrated the highest difference between the model and experiment is shown in Figure 1.9.

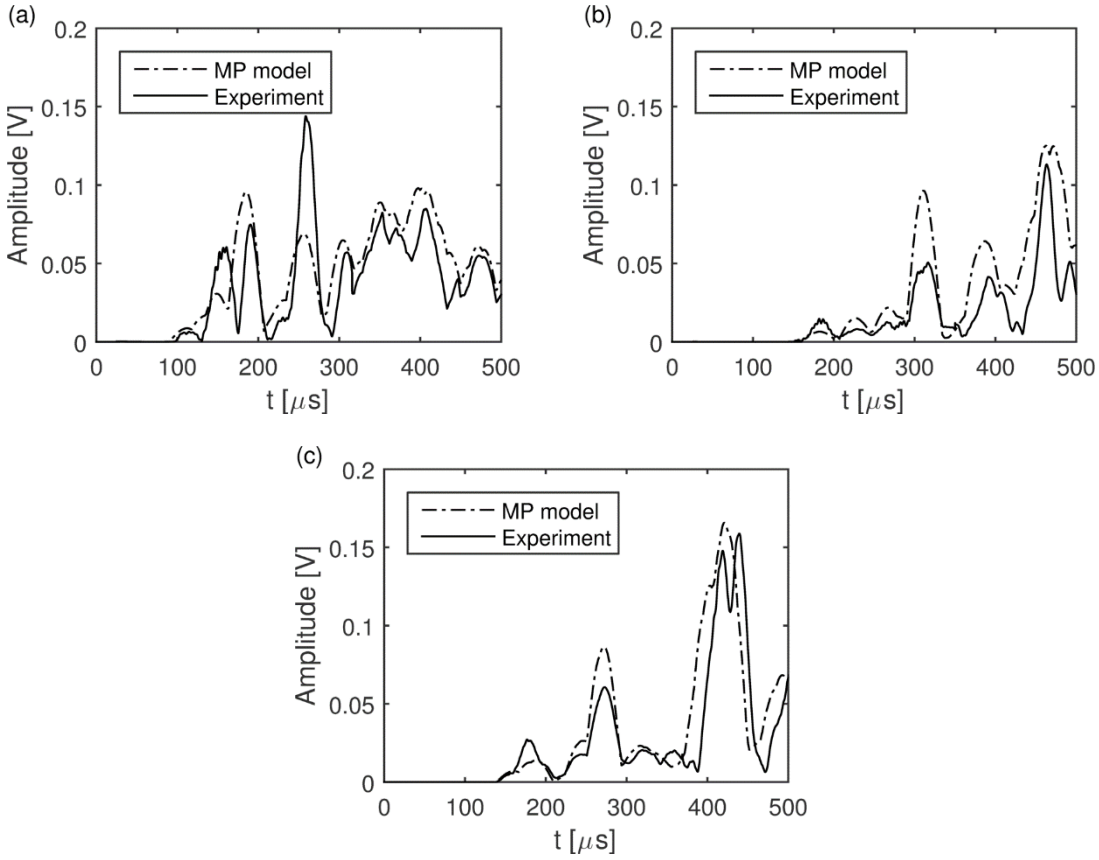

Figure 1.8: Envelope of scattered wave from M_1, MP model is compared with experimental data: (a) T_1-T_2, (b) T_1-T_3 (c) T_2-T_3.

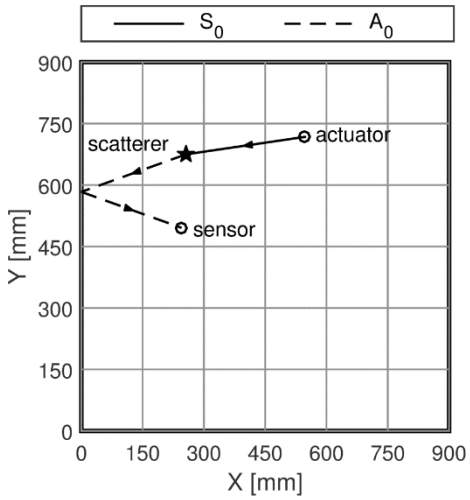

Figure 1.9: A path associated with the dominant peak of the Figure 1.8(a).

1.6.2. Imaging

The proposed MP imaging algorithm was compared with two other algorithms, that is, delay-and- sum (DS) (Wang et al. 2004) and minimum-variance (MV) (Hall and Michaels 2010). A brief overview of both algorithms is provided in section 1.11. The three imaging algorithms were applied to the same data set, and values of the reconstructed images were normalized to their maxima.

Figure 1.10 compares the images obtained by using the three imaging algorithms, for damage (magnets) located at M_1. Several artifacts can be observed in the images generated by the DS and MV algorithms (Figure 1.10(a-b)). Moreover, the amplitudes of some artifacts are higher than the actual damage location. In Figure 1.10(b), many low amplitude artifacts are suppressed by the MV algorithm, but still high amplitude artifacts exist in the image. On the other hand, Figure 1.10(c) shows that the maximum of the MP image is correctly detected close to the actual location of the damage. However, many artifacts exist in the image.

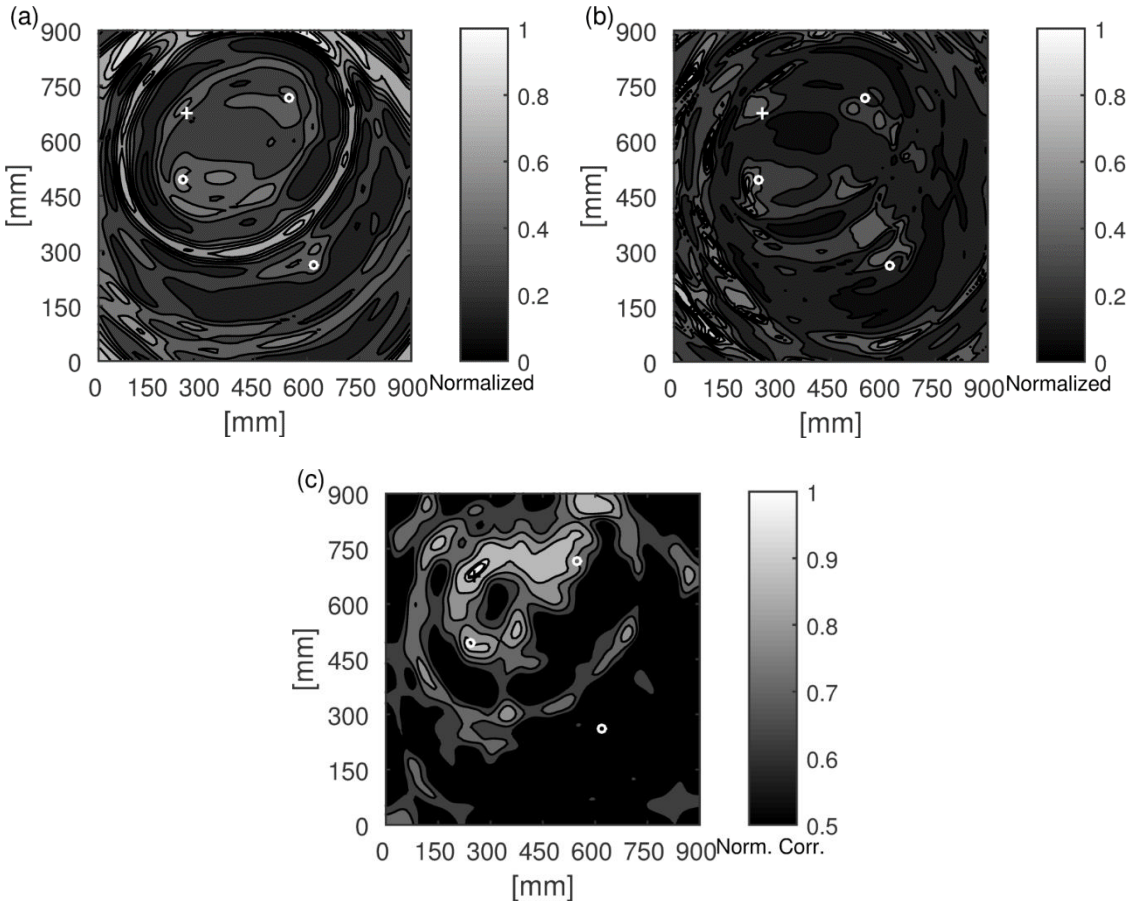

Figure 1.10: Image of magnets on M_1, actual location of damage is shown with a '+' sign, transducers are shown with white circles: (a) DS, (b) MV (c) MP.

Figure 1.11 shows images generated with a damage located at M_2. An overall better performance can be observed compared to the scenario in which damage was located outside the convex area identified by the transducers. However, DS and MV exhibit many artifacts outside the transducers area (Figure 1.11(a-b)). Although in the MV image low amplitude artifacts are suppressed, artifacts with higher pixel value than the actual damage are seen. In Figure 1.11(c), the maximum of the MP image is close to the location of the actual damage, but still some artifacts exists.

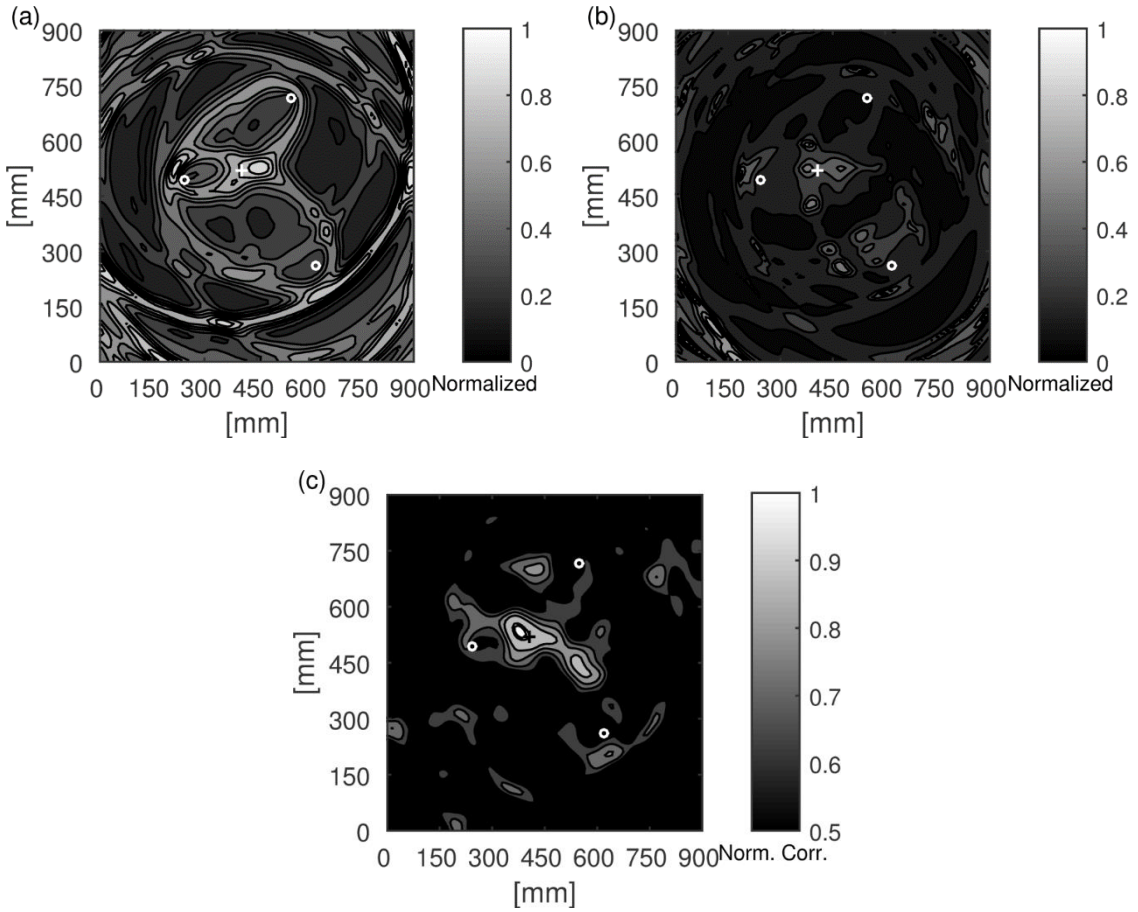

Figure 1.11: Image of magnets on M_2, actual location of damage is shown with a '+' sign, transducers are shown with white circles: (a) DS, (b) MV (c) MP.

Figure 1.12 shows images generated with a 2 mm through-thickness hole located at M_1. To test the applicability of approach in localizing progressive damages, baselines for this test were collected drilling a 1-mm hole at the same location. Also in this case the image generated by the MP approach exhibits an overall better performance compared to the DS and MV images.

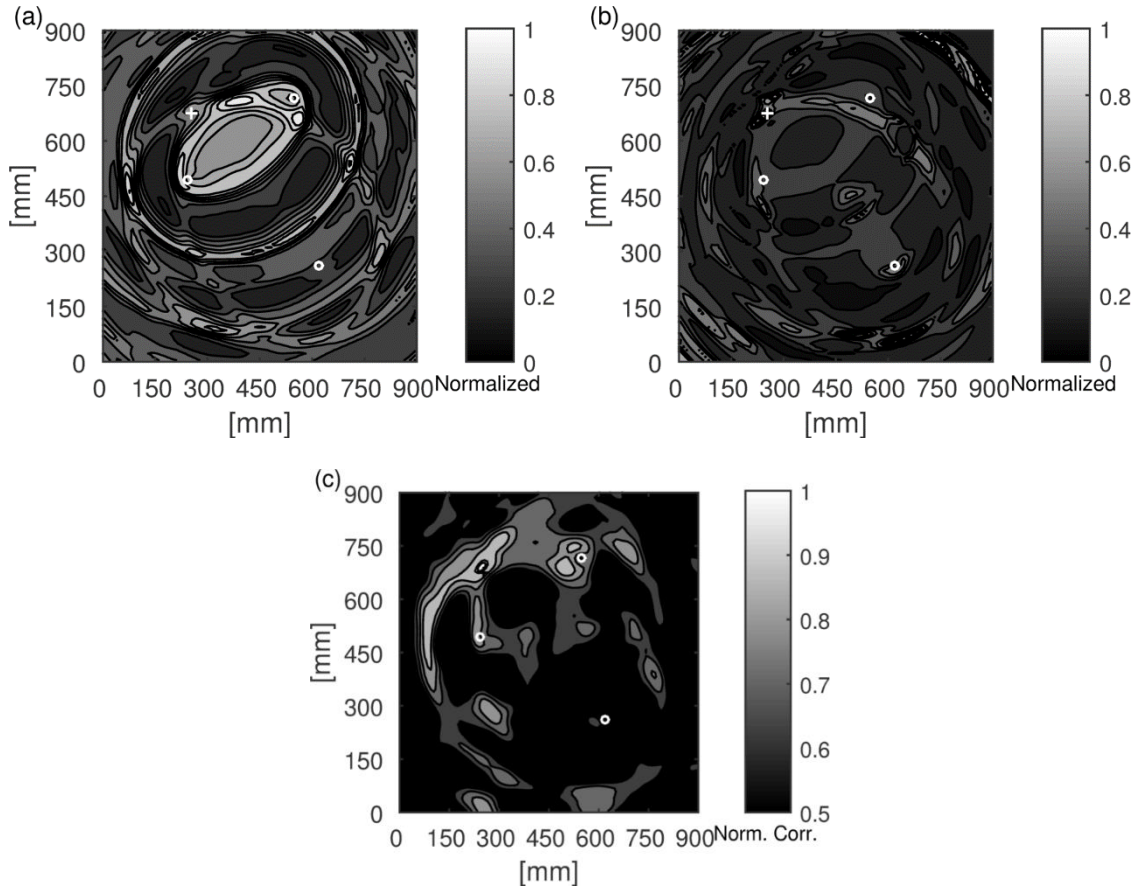

Figure 1.12: Image of a 2 mm through thickness drilled hole at M_1 ('+' sign), baseline was collected after drilling a 1-mm hole; transducers are shown with white circles: (a) DS, (b) MV, (c) MP.

1.6.3. Computation time

Computation time is the sum of the time required for: 1) envelope estimation, and 2) correlation imaging. Because envelope estimation is repeated for each pixel, it requires several calls to the model and takes longer to be computed. However, this step needs to be repeated only after a new baseline collection. In other words, the same estimated envelopes can be used multiple times for the routine GUW imaging. Given a set of estimated envelopes, a baseline signal, and a new GUW measurement, generating the correlation

image takes only a few seconds. Table 1.5 shows the computation time of the envelope estimation and correlation imaging for different numbers of transducers. The computations are based on 100 × 100 pixel resolution which is the same resolution used throughout the paper.

Table 1.5: Computation time for generating a 100 × 100 pixel image on a core i5 PC.

Number of Transducers	Estimation Time [min]	Imaging Time [sec]
3	16	3
4	30	5
5	52	8
6	78	10

1.7. SUMMARY AND CONCLUSIONS

This study introduced a method for leveraging edge reflections of GUWs to enable damage localization in isotropic plate structures. The method consists of two main components: 1) a multipath model capable of predicting the envelope of scattered GUWs, and 2) an image reconstruction algorithm. Some assumptions were made on the model to overcome some of the challenges associated with reflections, scattering, mode conversion, dispersion, and attenuation of Lamb waves. As a result some mismatch between the model and experimental data were observed. Experimental tests showed a very good performance of the image reconstruction algorithm, even in the case of pre-existing damage conditions. The accuracy of the proposed approach demonstrates its potential application in real-time SHM applications. However, the model is currently limited to simple rectangular plates. Multiple point-like defects are treated as one-at-a-time progressive damages. Extension of the idea of including edge reflections in a model-based damage localizer on more realistic plates with arbitrary geometries, stiffeners, and large holes is left for the future studies. In

addition, the capability to detect large damaged zones or simultaneously growing multiple point-like defects needs to be developed. More formal tests need to be conducted to verify the robustness of the approach for a variety of defect types.

1.8. ACKNOWLEDGMENTS

This work was supported by the National Science Foundation under Grant No. CMMI-1333506.

1.9. ADDITIONAL MATERIALS: BASELINE CORRECTION

In order to minimize the negative effect of systematic and random errors on imaging, the following corrections are applied on the baseline data (y_b). Systematic errors are resulted from sampling in time and amplitude, and can be corrected with a reference signal (y_d). The corrections are applied in two separate steps. The first step finds the time mismatch (τ_d) in the sampling of the baseline and the reference signal, and the second step, finds amplitude adjustment factor (β) to correct round off and random errors in the outcome of step one (y_c). In these corrections, F_s stands for sampling rate, and the final corrected signal is βy_c. It is assumed packet of the first arrival of the S_0 mode is not affected by damage, so isolated from latest collected data and used as reference signal.

The first arrival of a baseline signal propagates on the direct line connecting the corresponding pair of transducers. If no damage is present in the vicinity of this line, the first arrival of the damage signal is expected to have the same TOF and amplitude as the first arrival of the baseline signal. In this study, it is assumed that no damage is located on the lines connecting transducers pairs. Therefore, the first arrival portion of the damaged signal is used as a reference to correct the entire baseline signal. If the damage lies on the line of sight of a transducer pair, the difference between the damaged and baseline signals

becomes significant. In this case, the effects of the systematic and random noise on the scatted waves can be neglected, and the baseline correction is not needed.

1.9.1. Sampling Time Mismatch Restoration

$$y_c[i] = y_b[i] + (y_b[i+1] - y_b[i]) \cdot F_s \cdot \tau_d \tag{1.20}$$

$$\min_{\tau} \frac{1}{2} \sum_i (y_d[i] - y_c[i])^2 = \min_{\tau} \frac{1}{2} \sum_i (y_d[i] - y_b[i] - (y_b[i+1] - y_b[i]) \cdot F_s \cdot \tau_d)^2 \tag{1.21}$$

$$\frac{\mathrm{d}}{\mathrm{d}\tau_d} (\frac{1}{2} \sum_i (y_d[i] - y_b[i] - (y_b[i+1] - y_b[i]) \cdot F_s \cdot \tau_d)^2) = 0 \tag{1.22}$$

$$\sum_i (y_b[i+1] - y_b[i]) \cdot (y_d[i] - y_b[i] - (y_b[i+1] - y_b[i]) \cdot F_s \cdot \tau_d) = 0 \tag{1.23}$$

$$\tau_d = \frac{\sum_i y_b[i] \cdot (y_d[i] - y_b[i]) - \sum_i y_b[i+1] \cdot (y_d[i] - y_b[i])}{F_s \cdot (\sum_i y_b[i] \cdot (y_b[i+1] - y_b[i]) - \sum_i y_b[i+1] \cdot (y_b[i+1] - y_b[i]))} \tag{1.24}$$

1.9.2. Amplitude Difference Restoration

$$\min_{\beta} \frac{1}{2} \sum_i (y_d[i] - \beta y_c[i])^2 \tag{1.25}$$

$$\frac{\mathrm{d}}{\mathrm{d}\beta} (\frac{1}{2} \sum_i (y_d[i] - \beta y_c[i])^2) = \sum_i -y_c[i] \cdot (y_d[i] - \beta y_c[i]) = 0 \tag{1.26}$$

$$\beta = \frac{\sum_i y_d[i] \cdot y_c[i]}{\sum_i (y_c[i])^2} \tag{1.27}$$

1.10. ADDITIONAL MATERIALS: MP RAY TRACKING

Ray tracking is implemented in algorithm 1.1. The algorithm is capable of tracking all paths up to a specified path order; however, in this study only up to second order paths are tracked.

Algorithm 1.1: MP ray tracking finds all possible paths connecting two transducers with no more than specified maximum order of reflections.

algorithm MP **is**
  input: *a* (actuator coordinates), *r* (receiver coordinates), *bList* (boundaries), *maxOrder*
  output: *pathList*

  pathList ← **null**
  for each *order* **in** {1 , 2 , … , *maxOrder*}
    bSeqList ← list *order* long sequences from *bList,* not having immediately repeated members†
    for each *bSeq* in *bSeqList*
      path ← *a*
      isValid ← **true**
      (forward check)
      for each *reflection* **in** {1 , 2 , … , *order*}
        if sequence *bSeq* **is not** physically possible‡
          isValid ← **false**
          break
        m ← mirror *path*[end] with respect to *bList*[*bSeq*[*reflection*]]
        append *m* **to** the end of *path*
      append *r* **to** the end of *path*
      (backward check)
      if *isValid*
        for each *point* **in** {order, … , 2 , 1]}
          pt1← *path*[*point* + 1], *pt2*← *path*[*point* + 2], *b*← *bList*[*bSeq*[*point*]]
          if *pt1* **is not** visible†† for *pt2* through *b*
            isValid ← **false**
            break
          path[*point* + 1] ← intersection of the line connecting *pt1* and *pt2*, with *b*
      if *isValid*
        append *path* **to** *pathList*
  return *pathList*

† *bSeqList* is a list of lists, each of its members are a sequence with length of *order*. These sequences are combinations from boundaries, C_{order} (size(*bList*)) that does not have any immediately repeated members

‡ If a sequence is not physically possible, it requires crossing trails of boundaries in a sequence which is different from *bSeq*

†† If a point is visible from a second point through a boundary, line connecting the two points intersects the boundary in the boundary's physical extent

1.11. ADDITIONAL MATERIALS: EXISTING IMAGING ALGORITHMS

1.11.1. Delay and Sum (DS)

The DS method in matrix notation can be expressed as:

$$p_{x,y}^{\mathrm{DS}} = \boldsymbol{e}_{x,y}^{\mathrm{T}} \boldsymbol{R}_{x,y} \boldsymbol{e}_{x,y} \tag{1.28}$$

In this representation, a look direction vector, $\boldsymbol{e}_{x,y}$ (defined later) and a spatiotemporal correlation matrix $\boldsymbol{R}_{x,y}$ are used. The spatiotemporal correlation matrix is defined as:

$$\boldsymbol{R}_{x,y} = \int \boldsymbol{r}_{x,y}(t) \cdot \boldsymbol{r}_{x,y}^{\mathrm{T}}(t) \cdot W(t) \mathrm{d}t \tag{1.29}$$

which $W(t)$ is a windowing function. In this study a Gaussian window function with length of $2.5n_c / f_c$ is used (n_c is number of cycles in the excitation, and f_c is its central frequency). $\boldsymbol{r}_{x,y}(t)$ is a vector of back propagated signals from the scatterer at the coordinates (x, y). Elements of this vector are signals (r_p) from P transducer pairs:

$$\boldsymbol{r}_{x,y}(t) = [r_1(t - \frac{D_{1,1}}{c_g}) \quad r_2(t - \frac{D_{1,2}}{c_g}) \quad \cdots \quad r_P(t - \frac{D_{1,P}}{c_g})]^{\mathrm{T}} \tag{1.30}$$

where $\mathbf{D}$ is a $Q \times P$ matrix constructed as $\mathbf{D} = [\mathbf{d}_{p=1} \quad \mathbf{d}_{p=2} \quad \cdots \quad \mathbf{d}_{p=P}]$. To be consistent with the rest of the paper, vector $\mathbf{d}$ is defined as $\mathbf{d} = \mathbf{d}^{\mathrm{I}} + \mathbf{d}^{\mathrm{II}}$. Therefore, $D_{1,p}$ is the direct arrival distance between the transducers of pair p after being scattered at (x, y). The look direction vector $\boldsymbol{e}_{x,y}$ in Eq. (1.28) is defined as:

$$\boldsymbol{e}_{x,y} \sim [\frac{1}{\sqrt{D_{1,1}^{\times}}} \quad \frac{1}{\sqrt{D_{1,2}^{\times}}} \quad \cdots \quad \frac{1}{\sqrt{D_{1,P}^{\times}}}]^{\mathrm{T}} \tag{1.31}$$

where $\mathbf{D}^{\times}$ is a $Q \times P$ matrix constructed as $\mathbf{D}^{\times} = [\mathbf{d}_{p=1}^{\times} \quad \mathbf{d}_{p=2}^{\times} \quad \cdots \quad \mathbf{d}_{p=P}^{\times}]$, and vector $\mathbf{d}^{\times}$ is defined as $\mathbf{d}^{\times} = \mathbf{d}^{\mathrm{I}} \circ \mathbf{d}^{\mathrm{II}}$ which is the product of the scatterer distance to each of the transducers in the pair p.

1.11.2. Minimum Variance (MV)

The MV is a variation of the DS which uses optimized weights, $\boldsymbol{w}_{x,y}$, instead of the look direction to minimize the artifacts. This method in matrix notation can be expressed as:

$$p_{x,y}^{\mathrm{MV}} = \boldsymbol{w}_{x,y}^{\mathrm{T}} \boldsymbol{R}_{x,y} \boldsymbol{w}_{x,y} \tag{1.32}$$

where the optimal weights are defined as:

$$\boldsymbol{w}_{x,y} = \frac{\boldsymbol{R}_{x,y}^{-1} \boldsymbol{e}_{x,y}}{\boldsymbol{e}_{x,y}^{T} \boldsymbol{R}_{x,y}^{-1} \boldsymbol{e}_{x,y}} \tag{1.33}$$

Since the spatiotemporal correlation matrix is a singular matrix, diagonal loading with 0.1 to 0.01of first eigenvalue of the matrix can be used to make inversion possible.

$$\boldsymbol{R}_{x,y}^{-1} \approx (\boldsymbol{R}_{x,y} + 0.1\lambda_1 I)^{-1} \tag{1.34}$$

PAPER 2

Acoustic Emission Source Localization in Thin Metallic Plates: A Single-sensor Approach based on Multimodal Edge Reflections

Arvin Ebrahimkhanlou and Salvatore Salamone[1]

2.1. SYNOPSIS

This paper presents a new acoustic emission (AE) source localization for isotropic plates with reflecting boundaries. This approach that has no blind spot leverages multimodal edge reflections to identify AE sources with only a single sensor. The implementation of the proposed approach involves three main steps. First, the continuous wavelet transform (CWT) and the dispersion curves of the fundamental Lamb wave modes are utilized to estimate the distance between an AE source and a sensor. This step uses a modal acoustic emission approach. Then, an analytical model is proposed that uses the estimated distances to simulate the edge-reflected waves. Finally, the correlation between the experimental and the simulated waveforms is used to estimate the location of AE sources. Hsu-Nielson pencil lead break (PLB) tests were performed on an aluminum plate

[1]This chapter have been previously published as: Ebrahimkhanlou, Arvin, and Salamone, Salvatore (2017). "Acoustic emission source localization in thin metallic plates: A single-sensor approach based on multimodal edge reflections." Ultrasonics, 78, 134–145, DOI: https://doi.org/10.1016/j.ultras.2017.03.006
According to the publisher's website: https://www.elsevier.com/about/policies/sharing accessed on July 25, 2018 inclusion of accepted manuscripts in a research dissertation is permitted. In addition, the dissertation advisor has allowed the direct reproduction of this paper. This is a non-typeset version of the accepted manuscript except for the numbering of sections, figures, tables and equations that have been adapted from the original to satisfy the formatting requirements for a dissertation. The author of this dissertation was the lead investigator conduction the research. The author was responsible of developing theories, designing and conducting experiments, analyzing the results, and writing the journal paper. The coauthors contributed by providing editorial comments.

to validate this algorithm and promising results were achieved. Based on these results, the paper reports the statistics of the localization errors.

Keywords: source localization, guided ultrasonic waves, impact localization, modal acoustic emission, reverberations, structural health monitoring

2.2. INTRODUCTION

Plate-like structures are ubiquitous in civil, marine, and aerospace structures. Examples include bridge girders, aircraft wings and fuselages, ship hulls, and etc. (Kundu 2014; Yu et al. 2012). Corrosion, fatigue cracking, and impacts are some of the most common types of threats to these components. Since these defects are acoustic emission (AE) sources, several AE-based structural health monitoring (SHM) techniques have been developed in order to localize such defects. For example, Kundu et al. (2007) formulated an optimization-based solution for plates with known wave velocities. They later used clusters of three AE sensors to localize sources in anisotropic plates with unknown wave velocities (Kundu et al. 2012, 2015; Nakatani et al. 2014; Park et al. 2016). As another example, Niri et al. (2012; 2014) used the Kalman filter to develop a probabilistic source localization framework first for isotropic plates (Dehghan Niri and Salamone 2012) and then later for anisotropic plates (Dehghan Niri et al. 2014). Conventionally, these techniques use the first-arrival time of AE signals detected at multiple receiving points to locate the damage. Although this approach works relatively well for simple structures, realistic structures often have geometrical features, such as joints, stiffeners, and stringers that generate multiple acoustic reflections. These reflections could reduce the reliability of current source localization approaches in terms of automatic damage detection. One strategy typically used to overcome this challenge is to increase the number of sensors,

which can dramatically increase the complexity of the system and its deployment cost. A more effective alternative is to not only *account for* such reflections but also leverage the additional information that they convey to improve the localization accuracy. For instance, Achdjian et al. (2014) formulated a statistical multi-reflection model, which uses the propagated energies in the codas (tails) of at least three AE signals in localize their source. More recently, Ernst et al. (2016) proposed an approach to localize AE sources on a thin metallic plate by back propagating the edge-reflected late arrivals of the first antisymmetric Lamb wave mode (i.e. the A_0 mode). They used a finite element model (FEM) to back propagate the velocity signals collected from a single point laser Doppler vibrometer (LDV) and reported the required computation time for each AE localization to be six hours. They also discussed that placing the sensor on any of the symmetry lines of the plate creates symmetric wave fields that results in localization ambiguities. Moreover, Ciampa and Meo (2011) demonstrated the potential of using edge reflections for the single-sensor localization of AE sources. They developed a data-driven algorithm, which uses previously collected (i.e. baseline) wave-field data and correlation imaging to localize AE sources. Other than edge-reflection-based techniques, the modal acoustic emission is another family of the techniques that reduce the number of AE sensors in order to overcome the high costs associated with sensors and data acquisition channels (Jiao et al. 2004, 2008; Surgeon and Wevers 1999; Toyama et al. 2001). According to these techniques, the multimodal characteristics of AE signals in plate-like structures can be used to localize AE sources with only two sensors (Jiao et al. 2008). In this family of source localization techniques, Grabowski et al. (2016) recently developed a wavenumber-frequency mapping technique called Time-Distance Domain Transform that combines the triangulation technique with

the modal acoustic emission to increase the accuracy of the there-sensor triangulation algorithm.

Despite these notable contributions, still single-sensor source localization algorithms, even for simple metallic structures, require either excessive baseline collection or intensive computations. To overcome these challenges, this paper introduces a novel source localization algorithm that leverages the echoes and reverberations of multiple Lamb wave modes in AE signals. The scope of this algorithm includes all bounded structures that at least two edge reflections can be identified in their AE signals. The main idea is to develop a hybrid algorithm that effectively combines the modal acoustic emission and edge-reflection-based techniques. Such an algorithm has the advantages of both techniques. To prove the concept, a thin metallic plate with free edges is specifically considered. Although such free edges may not exist in all real-world structures (e.g., an airplane fuselage), reflections are expected from the stiffeners, stringers, and frames that divide the skin plate of those structures into surrounded panels. These reflections are due to the considerable stiffness change and thus wave velocities differences that occur at the boundaries of such panels. Therefore, the overarching goal in the future of this study is to monitor each of those panels with only one sensor.

The proposed algorithm consists of three key steps (see Figure 2.1). First, the arrival time measurements of both fundamental Lamb wave modes (i.e., S_0 and A_0) are conducted at various frequencies to estimate the distance between the AE source and the sensor (Step I). Then, an analytical model (hereafter referred to as the *multipath (MP) model*) is developed to simulate their late-arrival wave packets (Step II). Finally, a correlation imaging approach is used to localize the AE source (Step III).

The rest of the paper is organized as follows: section 2.3 introduces the source localization algorithm and discusses its theoretical aspects. Section 2.4 explains the experimental setup, and section 2.5 goes over the achieved source localization results, their accuracy, and computational cost. Finally, section 2.6 presents the concluding remarks. Two other sections containing additional materials also accompany the paper.

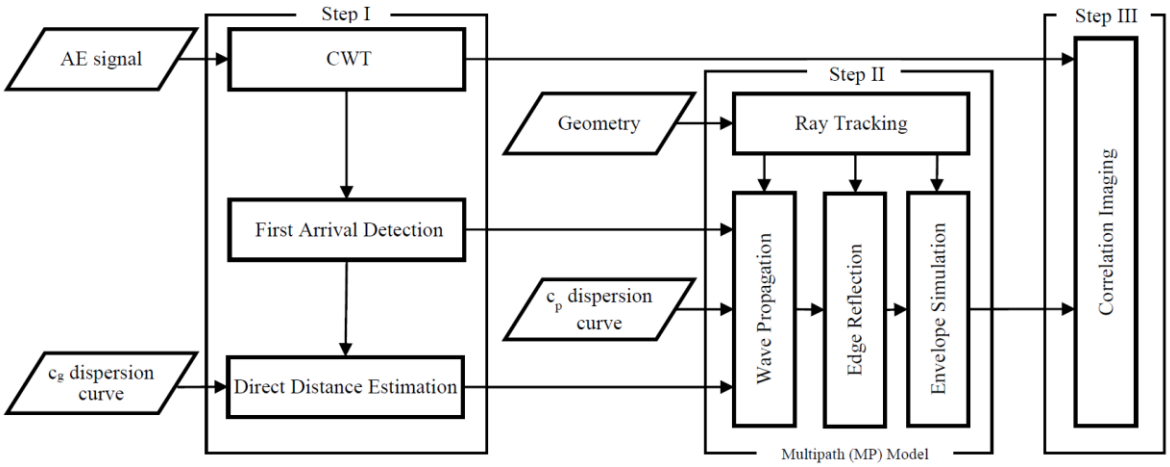

Figure 2.1: Flowchart of the proposed source localization approach.

2.3. SOURCE LOCALIZATION ALGORITHM

This section discusses the three necessary steps to implement the proposed approach.

2.3.1. Source-to-sensor distance estimation

Consider an AE source located at distance *d* from a sensor (see Figure 2.2(a)). To estimate *d*, first a continuous wavelet transform (CWT) is performed on the received AE signal as:

$$C_W(f,t_w)=\frac{1}{\sqrt{s_W(f)}}\int_{-\infty}^{+\infty}s(t)\Psi_W^*\left(\frac{t-t_W}{s_W(f)}\right)dt \qquad (2.1)$$

where $s_W(f)$ is the non-dimensional scale parameter, t_w is the translation parameter, and $\Psi^*_W(t)$ is the complex conjugate of the complex Morlet mother wavelet $\Psi_W(t)$, defined as (Mallat 1998):

$$\Psi_W(t) = \frac{1}{\sqrt{\pi f_b}} \exp(2\pi f_c \mathrm{j}t - \frac{t^2}{f_b}) \tag{2.2}$$

The non-dimensional parameters, f_b and f_c, are the bandwidth parameter and central frequency, respectively. The scale parameter in Eq. (2.1) is defined as:

$$s_W(f) = \frac{f_c \cdot f_s}{f} \tag{2.3}$$

where f_s is the sampling frequency of $s(t)$. The real part of the CWT could be interpreted as a Gaussian band-pass filter that has a central frequency and a standard deviation equal to f and $f/(2\pi f_c\sqrt{f_b})$, respectively (Dehghan Niri and Salamone 2012; Tse and Lai 2007); therefore, using the translation parameter as the time vector, the filtered signal could be represented as:

$$r(f,t) = \mathrm{Re}(C_W(f,t_w)) \tag{2.4}$$

For any Lamb wave mode in $r(f,t)$ the time of flight is inversely proportional to the group velocity, $c_g(f)$. This is because of the fact that the propagation distance, *d*, is the same for all frequencies and modes, that is:

$$\mathbf{c}_g \circ (\boldsymbol{\tau} - \tau_{AE}\mathbf{1}) = d\mathbf{1} \tag{2.5}$$

where τ_{AE} is the unknown time of the AE event, the vector $\boldsymbol{\tau}$ contains the time of the arrivals of the two fundamental modes (S_0 and A_0) at different frequencies, the vector $\mathbf{c_g}$ contains their corresponding group velocities, and $\mathbf{1}$ is a vector with all elements equal to one. The symbol ($\circ$) represents an element-wise product. To calculate the arrival time of S_0 and A_0, the Akaike information criterion (AIC) and a threshold-based approach are used, respectively (see section 2.8).

Defining $\mathbf{A}=\begin{bmatrix}\mathbf{c}_g & \mathbf{1}\end{bmatrix}$, $\mathbf{v}=\begin{bmatrix}\tau_{AE} & d\end{bmatrix}^T$, and $\mathbf{b}=\mathbf{c}_g \circ \boldsymbol{\tau}$, Eq. (2.5) could be rearranged as a system of equations:

$$\mathbf{Av}=\mathbf{b} \tag{2.6}$$

Since the number of equations is higher than the number of two unknowns (i.e. τ_{AE} and d), the system of equations is overdetermined. The least squares (LS) method is used to solve Eq. (2.6):

$$\mathbf{v}=(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b} \tag{2.7}$$

When $\boldsymbol{\tau}$ is measured for both the S_0 and A_0 modes, at least two non-parallel equations exists in $\mathbf{A}$ ($\det(\mathbf{A}^T\mathbf{A})\neq 0$). However, when only one mode is considered, the frequency f should be sampled from the dispersive range of that mode; the matrix $\mathbf{A}^T\mathbf{A}$ otherwise approaches to the singularity.

2.3.2. Multipath (MP) model

The MP model is an analytical model that simulates edge-reflected wave packets. This model uses the first arrivals of filtered AE signals to reconstruct their late-arrival packets. To perform this task, the model uses four modules: ray tracking, wave propagation, edge reflection, and envelope simulation. The following subsections discuss each module in details.

2.3.2.1. Ray tracking

Ray tracking is a common technique to calculate the propagation path of waves through a medium. For instance, many studies have used this technique to track Lamb waves in plate-like structures (Ebrahimkhanlou et al. 2016b; Harley and Moura 2013; Levine and Michaels 2013; Muller et al. 2016). However, to the best of the authors' knowledge, all such applications have been mainly in the *active* mode, where the Lamb

waves are excited by an actuator, rather in the *passive* AE mode. For example, the authors have developed the multipath (MP) ray-tracking algorithm in their previous publication (Ebrahimkhanlou et al. 2016b). This algorithm considers multiple edge reflections and traces the propagation paths of the Lamb waves from an actuator to a scatterer (damage) and finally to a sensor. In this paper, we propose using a modified version of the MP ray-tracking algorithm to trace the propagation paths of AE signals from a source to a sensor. Therefore, this algorithm is reviewed and adopted to trace the propagation paths of AE signals.

The propagation paths from an AE source to a sensor are either the *direct path* or one of the many *indirect paths*. The direct path is the commonly depicted line-of-sight (i.e., the straight line) between the source and the sensor. An indirect path is a path that ends at the sensor after one or several reflections from the edges of the plate. In order to calculate the propagation paths, the MP ray-tracking algorithm needs the following parameters: (i) the dimensions of the plate, (ii) the coordinates of the sensor, (iii) an initial guess for the coordinates of the AE source, and (iv) the maximum number of traced reflections on a path, $o_{\max}$. The algorithm calculates all possible paths that satisfy this maximum number. In a frequency range below the first cutoff frequency of the Lamb waves, the only propagating modes are the first symmetric (S_0), antisymmetric (A_0), and shear horizontal (SH_0) modes. At the edges of the plate, an incident S_0 mode reflects as S_0 and SH_0, whereas an incident A_0 mode reflects only as an A_0 without any mode conversion (Le Clezio et al. 2003; Gunawan and Hirose 2007; Perelli et al. 2012; Torvik 1967). The SH_0 mode is not considered in this study because the AE sensors used in the experiments have a negligible sensitivity to this mode. To calculate the propagation paths, Snells law is used. Snell's law

governs the relation between the incident and reflection angles (Gunawan and Hirose 2007):

$$k_{\mathrm{I}} \sin(\theta_{\mathrm{I}}) = k_{\mathrm{R}} \sin(\theta_{\mathrm{R}}) \tag{2.8}$$

where k_{I} and θ_{I} are, respectively, the incident wave's wavenumber and the incident angle. Similarly, k_{R} and θ_{R} are, respectively, the wavenumber and angle of the reflected wave (see Figure 2.2(c)). Without any mode conversion, the wavenumber of the incident and reflected waves are the same. Consequently, Eq. (2.8) requires equal incident and reflected angles. In another word, the edges of the plate act as mirrors if no mode conversion occurs. Figure 2.2 visualize the overall procedure used to calculate the propagation paths from an arbitrary source to a sensor. Specifically, Figure 2.2(a) shows the only direct path; Figure 2.2(b) shows one of the indirect paths with only a single edge reflection. To calculate this path, the source is first mirrored with respect to a reflecting edge. The figure shows the reflected source in one of the gray areas, which are the mirrored versions of the plate with respect to the four reflecting edges. Then, the line that connects the sensor and the mirrored source is considered. This line defines the propagation path until it intersects one of the edges. Finally, the intersection point is connected to the initial source location (i.e. its location before the mirroring) to track the rest of the path. Figure 2.2(c) shows the generalization of this procedure for one of the indirect paths containing two reflections from the plate's edges. Further implementation details of the MP ray-tracking algorithm could be found in the previous work of the authors (Ebrahimkhanlou et al. 2016b).

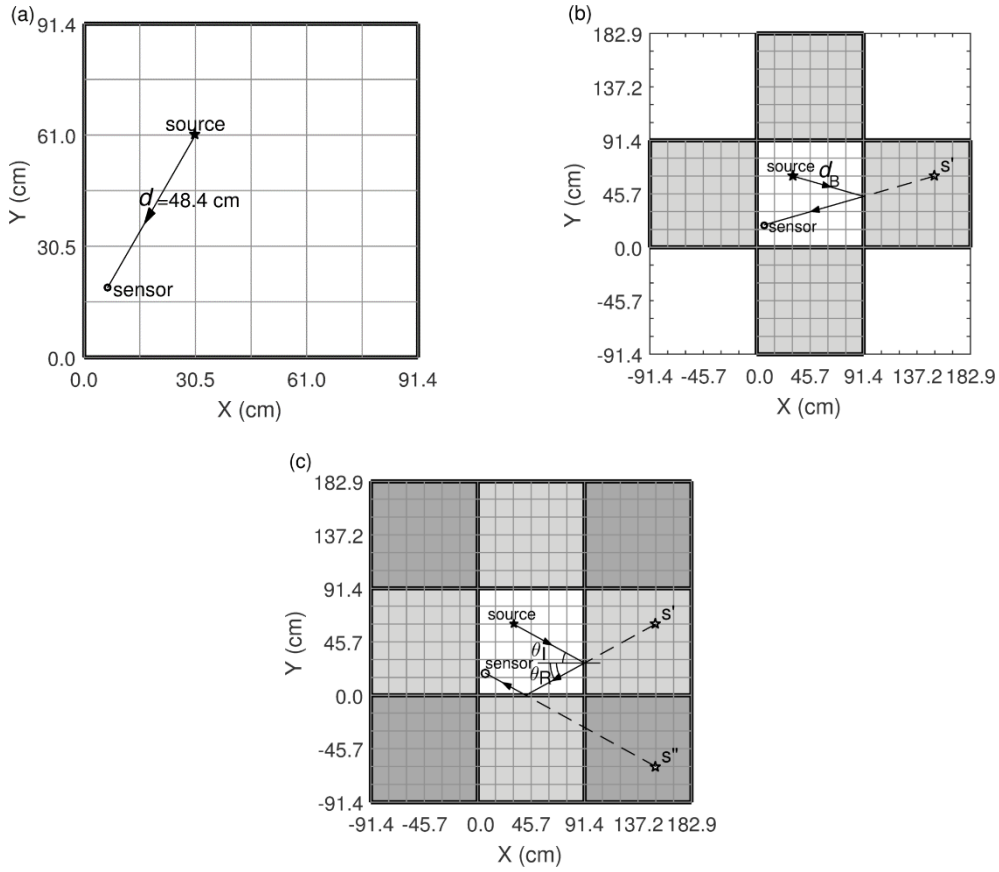

Figure 2.2: The intermediate steps of the MP ray-tracking algorithm: (a) a direct path, (b) a path with one reflection, (c) a path with two reflections.

The MP ray-tracking algorithm provides all possible paths connecting an AE source to a sensor. Theoretically, there are an infinite number of such paths. Therefore, the number of reflections that can occur on each path is limited to the maximum number, $o_{\max}$, specified as an input to the algorithm. Only the paths that satisfy this condition are considered. The number of such paths is defined as parameter q. The algorithm sorts these paths in the order of their lengths. Therefore, the first path is always the direct line connecting the source to the sensor (see Figure 2.2(a)). For the *i*-th path, the algorithm

returns the length of that path, d_i, and the number of reflections that occur on it, o_i. According to this notation, $d_1 = d$, where *d* was defined in section 2.3.1, and $o_1 = 0$.

2.3.2.2. Wave propagation

Given a function $u_0(t)$ for the out-of-plane displacement of the plate at the source, the out-of-plane displacement of an excited Lamb wave mode at a distance d_i from the source could be calculated as (Wilcox 1998):

$$u(d_i,t) = \mathcal{F}^{-1}\{\mathcal{F}\{u_0(t)\}E(k,\omega)H_0^{(1)}(kd_i)\} \tag{2.9}$$

where $E(k,\omega)$ is the excitability function of the considered mode; $H_0^{(1)}(\cdot)$ is the zero-order Hankel function of the first kind; d_i is the propagation distance of the *i*-th arrival; *k* is the wavenumber of the considered mode; and ω is the angular frequency. $\mathcal{F}\{\cdot\}$, and $\mathcal{F}^{-1}\{\cdot\}$ are the Fourier transform and its inverse, respectively. Although Eq. (2.9) is valid for any Lamb wave modes, each mode needs to be considered separately. One challenge with the AE sources is that the out-of-plane displacement at the source (i.e. $u_0(t)$) is unknown. To overcome this limitation, Eq. (2.9) is rearranged in terms of the direct source-to-sensor first arrivals of each mode:

$$\mathcal{F}\{u_0(t)\}E(k,\omega) = \frac{\mathcal{F}\{u(d_i,t)\}}{H_0^{(1)}(kd_i)} \tag{2.10}$$

Evaluating Eq. (2.10) at $d_1 = d$ (i.e. the direct distance from the source to the sensor) and substituting its left-hand side into Eq. (2.9) yields:

$$u(d_i,t) = \mathcal{F}^{-1}\{\mathcal{F}\{u(d,t)\}\frac{H_0^{(1)}(kd_i)}{H_0^{(1)}(kd)}\} \tag{2.11}$$

where $u(d,t)$ is the first arrival of the considered mode. For not-close-to-zero input arguments, the Hankel function could be approximated as (Abramowitz and Stegun 1965):

$$H_0^{(1)}(kd_i) = \sqrt{\frac{2}{\pi kd_i}}\exp(\mathrm{j}kd_i - \frac{\mathrm{j}\pi}{4}) \tag{2.12}$$

where j is the imaginary unit ($\sqrt{-1}$). Substituting Eq. (2.12) into Eq. (2.11) and using $k = 2\pi f / c_{\mathrm{p}}(f)$:

$$u(d_i,t) = \mathcal{F}^{-1}\{(\frac{d_i}{d})^{-0.5}\mathcal{F}\{u(d,t)\}\exp(\frac{\mathrm{j}2\pi f(d_i-d)}{c_{\mathrm{p}}(f)})\} \tag{2.13}$$

where c_{p} is the phase velocity dispersion curve of the considered mode.

Therefore, given d from the direct distance estimation, Eq. (2.13) propagates a first arrival, $u(d,t)$, to a distance d_i from the source. The MP ray-tracking algorithm provides the distance d_i (see section 2.3.2.1). To identify the first-arrival packet (i.e. $u(d,t)$), first arrival isolation methods are proposed in section 2.8. These methods, which are applied to the real part of the CWT coefficient (i.e. $r(f,t)$), return the first S_0 and A_0 packets.

2.3.2.3. Edge reflection

The edge reflected Lamb waves could be calculated as (Ernst et al. 2016):

$$u_{\mathrm{R}}(d_{\mathrm{B}},t) = (\gamma)\mathcal{F}^{-1}\{\mathcal{F}\{u_{\mathrm{I}}(d_{\mathrm{B}},t)\}\exp(\mathrm{j}\varphi)\} \tag{2.14}$$

where $u_{\mathrm{I}}(d_{\mathrm{B}},t)$ is the incident wave; γ is the attenuation coefficient (Ebrahimkhanlou et al. 2016a); φ is the phase-shift; and d_{B} is the distance from the source to the reflecting boundary (see Figure 2.2(b)). The late arrivals of each mode are calculated by combining the Eq. (2.13) and Eq. (2.14):

$$u(d_i,t) = \gamma^{o_i}\mathcal{F}^{-1}\{(\frac{d_i}{d})^{-0.5}\mathcal{F}\{u(d,t)\}\exp(\frac{\mathrm{j}2\pi f(d_i-d)}{c_{\mathrm{p}}(f)} + \mathrm{j}\varphi')\} \tag{2.15}$$

where φ' is the overall phase-shift due to the reflections occurred on a path. The values of d_i and o_i are determined from the MP ray-tracking algorithm for a guessed source coordinates **x**. In this study, φ' is assumed to be frequency independent (i.e. it shifts the wave without distorting it).

2.3.2.4. Envelope simulation

Envelope simulation sums the edge-reflected wave packets to reconstruct the envelope of filtered AE signals. These filtered signals are the sum of several S_0 and A_0 wave packets that have propagated through multiple paths. Therefore, the envelope of a filtered signal could be reconstructed as:

$$\mathbf{e}(\mathbf{x}) = \left| \sum_{i=1}^{q} (u_{S_0}(d_i,t) + u_{A_0}(d_i,t)) \right| \tag{2.16}$$

where $u_{S_0}(d_i,t)$ and $u_{A_0}(d_i,t)$ are respectively the i-th late arrivals of the S_0 and A_0 modes that come from a source located at the coordinates $\mathbf{x}$, the notation $|\cdot|$ indicates the modulus of the signal, and q is the total number of paths in the MP ray-tracking algorithm (see section 2.3.2.1). To calculate $u_{S_0}(d_i,t)$ and $u_{A_0}(d_i,t)$, Eq. (2.15) is used with the corresponding first arrivals, $u(d_1,t)$, phase velocities, $c_p(f)$, and attenuation coefficients, γ, for the S_0 and A_0 modes. However, the phase-shift, φ', is unknown in Eq. (2.15). Although φ' is too small to affect each individual arrival packet, it can change the constructive or destructive effects of the packets on each other. To eliminate the unknown phase-shift φ' without neglecting its effects, the square root of the sum of the squares (SRSS) is used instead of Eq. (2.16):

$$\mathbf{e}(\mathbf{x}) = \underset{i=1}{\overset{q}{\mathrm{SRSS}}}(u_{S_0}(d_i,t), u_{A_0}(d_i,t)) \tag{2.17}$$

2.3.3. Correlation imaging

Correlation imaging is a dictionary-based algorithm, which compares the similarity of the experimental and simulated signals in time domain to find the most similar simulation to the experiment (Ebrahimkhanlou et al. 2016a; b; Park et al. 2012; Quaegebeur et al. 2011). In a correlation image, the coordinates of the pixels **x**, are the initial guesses for simulating a source. According to this technique, the correlation $\rho(\mathbf{x})$, is assigned as the

value of the pixel located at **x**. The pixel with the highest correlation value is the estimated source location. The correlation $\rho(\mathbf{x})$, is defined as:

$$\rho(\mathbf{x}) = \frac{\sum_{i=1}^{n_s}(e_i(\mathbf{x}) - \bar{e}(\mathbf{x}))(e_i - \bar{e})}{\sqrt{\sum_{ii=1}^{n_s}(e_{ii}(\mathbf{x}) - \bar{e}(\mathbf{x}))^2 \sum_{iii=1}^{n_s}(e_{iii} - \bar{e})^2}} \tag{2.18}$$

where vector **e** is the envelope of a filtered experimental signal, the vector $\mathbf{e}(\mathbf{x})$ is the envelope of the simulated signal that comes from a source at the coordinates **x** (see Eq. (2.17)), and n_s is the length of $\mathbf{e}(\mathbf{x})$ (and also $\mathbf{e}$). To calculate **e**, the modulus of the CWT coefficients (i.e. $|C_W(f, t_w)|$) is used, where the frequency f is the same frequency used to simulate $\mathbf{e}(\mathbf{x})$. The bar on the quantities specifies their expected value (i.e. arithmetic mean):

$$\bar{e} = \frac{1}{n_s}\sum_{i=1}^{n_s} e_i \tag{2.19}$$

2.4. EXPERIMENTS

To validate the proposed source localization algorithm, experiments were performed on a 91.4 cm x 91.4 cm x 0.318 cm aluminum plate. To support the weight of the plate, four pieces of soft foam were placed under the corners. More details about the properties of the specimen could be found in Table 1.3. To simulate AE source, Hsu–Nielsen (Hsu 1977) pencil lead break (PLB) tests were performed on the specimen at the 64 points shown in Figure 2.3. Specifically, a 0.3mm mechanical pencil with 2H leads was placed at a 45-degree angle with respect to the plate, and its 3-mm-protruded lead was broken. To evaluate the localization error for the AE events that may occur at the same location, each PLB test was repeated four times. Therefore, an overall number of 256 PLB tests were performed. A broadband AE sensor (Physical Acoustics PICO) located at

coordinates (6.4 cm, 19.1 cm) was used to measure the AE signals. To avoid ambiguities in the localization results (see (Ernst et al. 2016)), these coordinates were selected in such a way that they do not intersect with any of the four symmetry lines of the plate (one horizontal, one vertical, and two diagonals). Placing a sensor on any of these lines creates symmetric correlation images that make it impossible to distinguish between the actual source and its mirrored version(s). A data acquisition (DAQ) system (Mistras Micro Express) digitized the AE signals after 40dB amplification (Physical Acoustics 2/4/6 preamplifier). The sampling rate was 2 MHz, and the low pass and high pass analog filters of the DAQ system were respectively set at 20 kHz and 400 kHz. AE signals were post-processed in MATLAB, and the dispersion curves of the plate (group velocities and phase velocities) were numerically calculated by solving the Rayleigh-Lamb equations (such curves could be found in (Rose 2004)).

Table 2.1: Properties of the tested plate.

Properties	Value
Material	Aluminum alloy 6061-T6
Dimension [cm × cm × cm]	91.4 × 91.4 × 0.318
Modulus of elasticity [GPa]	69
Poisson's ratio	0.33
Density [Kg/m^3]	2700

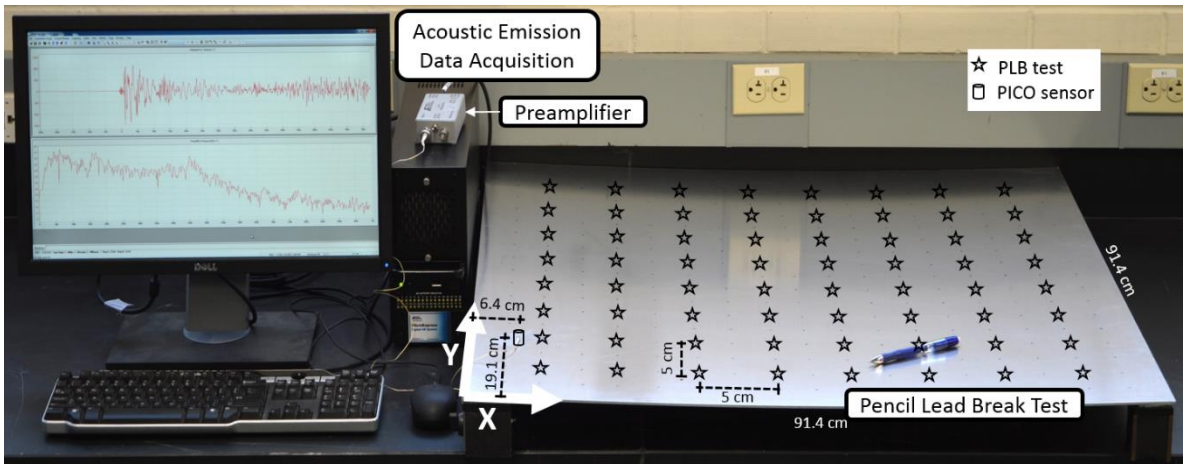

Figure 2.3: Experimental setup.

2.5. RESULTS AND DISCUSSION

This section presents and discusses experimental results for the intermediate steps and the overall performance of the proposed source localization algorithm. First, an AE signal generated by a PLB test is used to illustrate and validate the proposed source-to-sensor distance estimation (step I). Then, the MP simulations are discussed and compared with the same experimental signal (step II). Next, correlation imaging results are presented for three PLB tests (step III). Finally, the last two subsections use the average of the 256 PLB tests to discuss the overall performance of the proposed algorithm in terms of accuracy and computation time, respectively.

2.5.1. Source-to-sensor distance estimation

Figure 2.4 shows the AE signal used to validate the source-to-sensor distance estimation technique. One of the PLB tests performed at the coordinates (30.5 cm, 61.0 cm) was used to generate this wideband and multimodal signal. As shown in the figure, the reference time (i.e. the time zero) of the signal was defined as the trigger time.

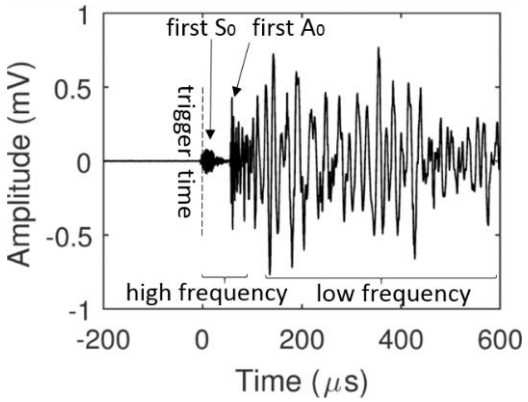

Figure 2.4: AE signal generated by one of the PLB tests at the coordinates (30.5 cm, 61.0 cm).

Figure 2.5(a-b) visualize the CWT of the AE signal shown in Figure 2.4. The non-dimensional bandwidth and central frequency parameters of the CWT were $f_b = 0.5$ and $f_c = 5$, respectively. Figure 2.5(a) shows the modulus of the CWT coefficients for a frequency vector **f** that was uniformly sampled from 25 kHz to 425 kHz every 1 kHz. This vector was selected based on the frequency response spectrum of the AE sensor and the analog filters used during the data acquisition process. It could be seen in the figure that the A_0 mode, which is the dominant mode, has a higher amplitude at the lower frequencies. In addition, the dispersion of this mode and its multiple reflections could be seen in the figure.

Figure 2.5(b) shows the real part of the CWT coefficients at the 75, 175, 275, and 375 kHz frequencies. These frequencies were selected among the frequencies listed in the vector **f** for the visualization purpose only. The first arrivals of the fundamental Lamb wave modes and several reflections of them are identified in the figure. It could also be seen that the lower the frequencies, the more delayed the first A_0 arrivals. According to the dispersion curves of the plate (see (Rose 2004) for the details of the procedure used to calculate the dispersion curves), this is because the corresponding group velocities for the

A_0 mode are 2476.3, 2972.3, 3090.3, 3112.5, and 3160.2 m/s, respectively. In addition, the figure shows that the A_0 mode has a higher amplitude than the S_0 mode. The lower the frequencies, the higher the amplitude of the A_0 mode, and the lower the amplitude of the S_0 mode. This was to an extent that the arrival time of the S_0 mode was not measurable at the frequencies less than 250 kHz.

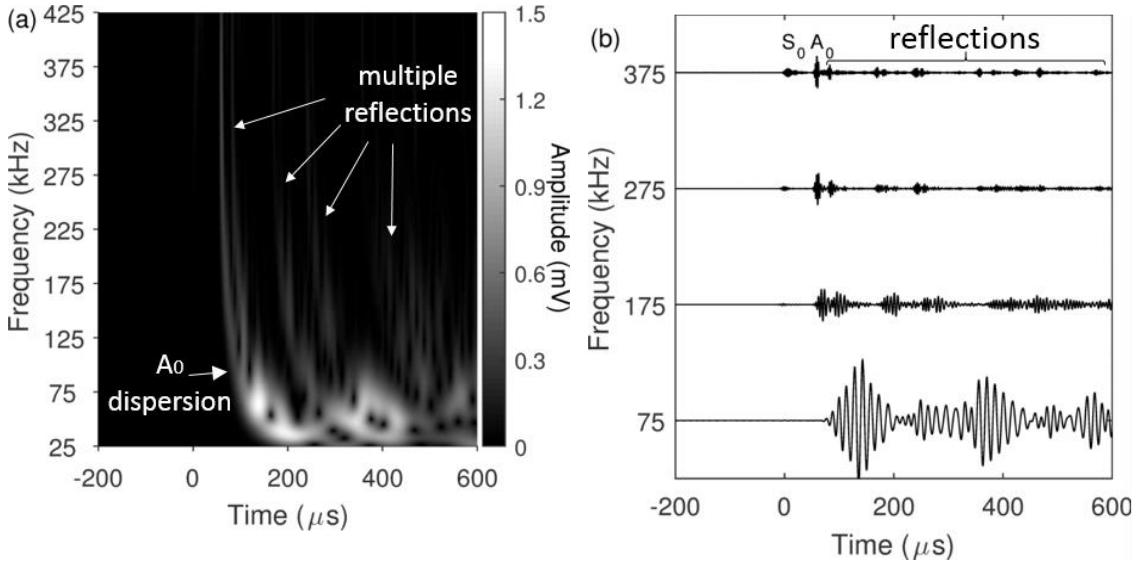

Figure 2.5: The CWT of a PLB: (a) the modulus of the CWT, (b) the real part of the CWT at the 75, 175, 275, and 375 kHz frequencies.

Figure 2.6 shows the first-arrival S_0 and A_0 packets and their arrival time (respectively τ_{S_0} and τ_{A_0}) for the real part of the CWT coefficients at 250 kHz (i.e. $r(f,t)|_{f=250\text{ kHz}}$). The filtered signal is shown in the background, and the first-arrival packets are highlighted. To isolate the first-arrival packets and measure their time of arrivals, the techniques presented in section 2.8 were used. In addition, the figure shows the corresponding time to the AE event (i.e. τ_{AE}), which was estimated by solving Eq. (2.5). As the figure shows, τ_{AE} is defined with respect to the trigger time and thus is always a negative number.

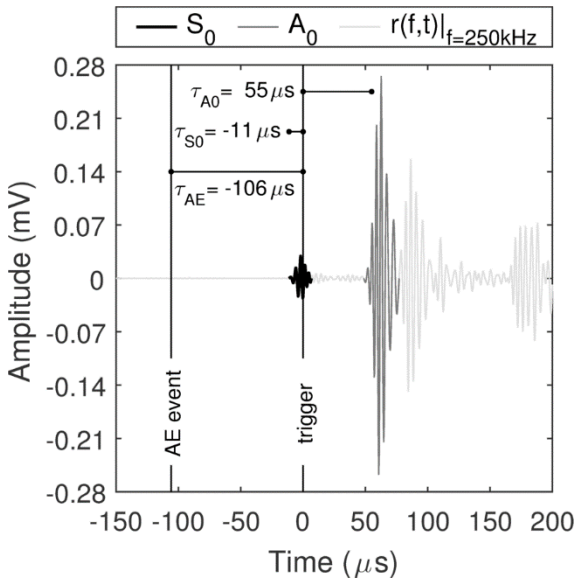

Figure 2.6: A filtered AE signal at 250 kHz; the first S_0 and A_0 packets, their arrival time, and the estimated time of the AE event are shown.

Figure 2.7 shows the measured S_0 and A_0 first-arrival time from the real part of the CWT coefficients. The first arrivals of the S_0 mode were measured at those frequencies listed in the vector **f** that were greater than 250 kHz because the S_0 mode had very low amplitudes at lower frequencies (the vector **f** was uniformly sampled from 25 kHz to 425 kHz at every 1 kHz). For a similar reason, the first arrivals of the A_0 mode were measured for the below 250 kHz frequencies in the vector **f**. These measurements were stored in vectors $\boldsymbol{\tau}_{S_0}$ and $\boldsymbol{\tau}_{A_0}$, respectively. In addition, the corresponding group velocities of the two modes were calculated from the dispersion curves of the plate and stored in vectors $\mathbf{c}_{gS_0}$ and $\mathbf{c}_{gA_0}$, respectively. To calculate the dispersion curves, the Rayleigh-Lamb equations were solved according to the numerical solution detailed in (Rose 2004). Then, the concatenations of the arrival time vectors (i.e. vector $\boldsymbol{\tau} = [\boldsymbol{\tau}_{S_0}^T, \boldsymbol{\tau}_{A_0}^T]^T$) and the group velocity vectors (i.e. $\mathbf{c}_g = [\mathbf{c}_{gS_0}^T, \mathbf{c}_{gA_0}^T]^T$) were used to construct Eq. (2.5). The estimated source-to-sensor distance and the occurrence time of the AE event were $d = 48.8$ cm and

$\tau_{AE} = -105.7\ \mu s$, respectively (the actual distance was 48.4 cm). To validate the solution, the vector $\boldsymbol{\tau}$ was assumed unknown. Then, given the estimated values for d and τ_{AE}, Eq. (2.5) was solved for $\boldsymbol{\tau}$. Figure 2.7 also shows these estimated values for the vector $\boldsymbol{\tau}$ and demonstrates their agreement with the measured values.

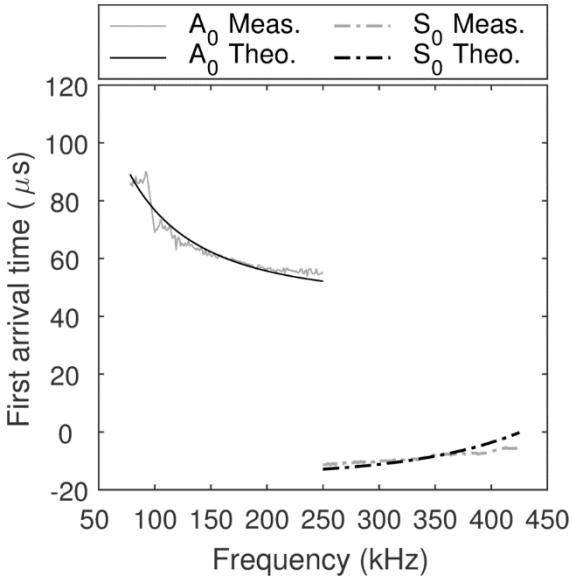

Figure 2.7: Comparison of the measured and estimated values for the time of first arrivals.

2.5.2. Multipath (MP) model

Figure 2.8 visualizes the output of the MP ray-tracking algorithm for the source of the AE signal shown in Figure 2.4. Twenty-five paths were calculated that three or fewer reflections occur on them (i.e. $q = 25$ and $o_{max} = 3$). For the sake of the figure's clarity, only some of the paths are shown. For each path, the travel distance d_i, and the number of reflections o_i, are shown. In addition, the detailed text output of the MP ray-tracking algorithm is presented in section 2.9.

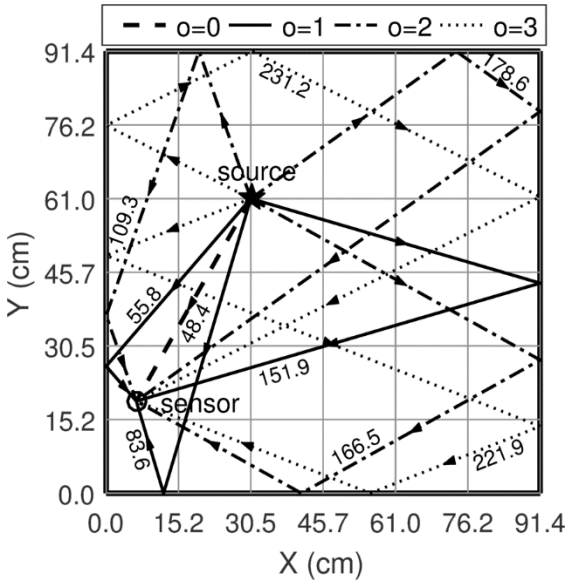

Figure 2.8: The output of the MP ray-tracking algorithm for up to three reflections; the length of each path is also included in centimeters.

Figure 2.9 shows the output of the wave propagation model for the isolated A_0 mode in Figure 2.6. Time shift, attenuation, and dispersion could be seen in the figure. To simulate the propagated packets, Eq. (2.13) was evaluated at additional 25, 50, 75, 100, and 125 cm propagation distances (i.e. $d_i - d$). In this equation, the estimated value for the direct source-to-sensor distance, d, was used (i.e. $d = 48.8\ \text{cm}$).

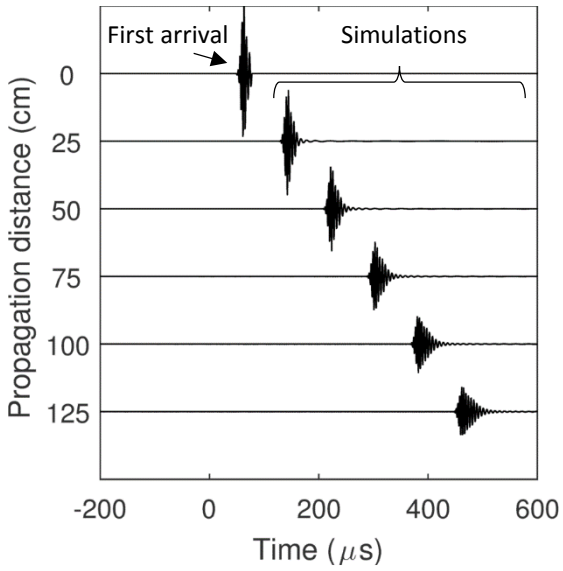

Figure 2.9: Wave propagation simulations; late arrivals are reconstructed from their first-arrival packets; the propagation distance is defined as $d_i - d$.

Figure 2.10 compares the experimental and simulated envelopes of the filtered signal shown in Figure 2.6. The experimental envelope is the modulus of the CWT coefficients at 250 kHz, $\mathbf{e}$, and the simulated envelope is the output of the MP model, $\mathbf{e}(\mathbf{x})$, for the actual source location (i.e. $\mathbf{x} = (30.5 \text{ cm}, 61.0 \text{ cm})$). The correlation between the two envelopes was 90.5 percent, which indicates the MP model can reconstruct late-arrival packets from their first arrivals. The slight differences between the two envelopes could be due to imperfections in the profile of the edges, which were assumed perfectly square cut as well as the supports of the plate, which were not modeled in the simulations. In these simulations, $\gamma_{S_0} = 0.5$ and $\gamma_{A_0} = 0.8$ were used for the reflections of the S_0 and A_0 modes, respectively. Although the simulations are not sensitive to the attenuation coefficients, a higher energy loss was assumed for the S_0 reflections to compensate for the mode conversion of the S_0 mode to SH_0.

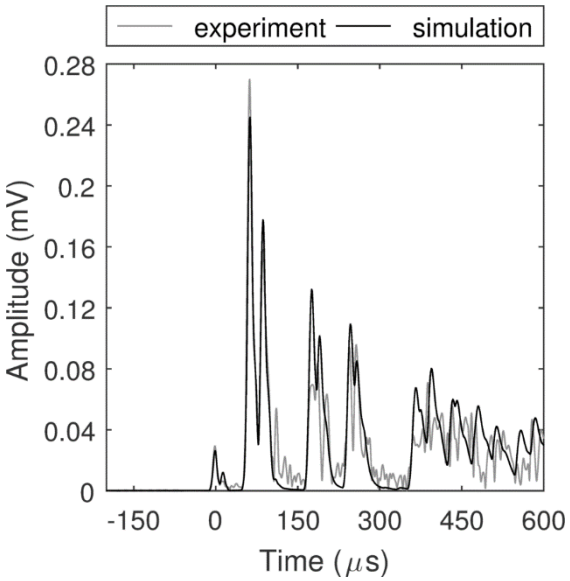

Figure 2.10: Comparison between the experimental and simulated envelopes of the signal shown in Figure 2.6.

2.5.3. Correlation imaging

Figure 2.11 shows correlation images for three PLB tests. The actual and estimated source locations could be seen in the figure. The highest correlation values are mainly located on an arc with the AE sensor at its center. In all three cases, such arcs cross the actual sources. As a results, the high-value pixels are less distributed in the radial direction (i.e. the direction of the source-to-sensor line) than the tangential direction (i.e. perpendicular to the radial direction). This is because the estimated source-to-sensor distance is embedded in the multipath (MP) simulations. Therefore, minimal correlation is expected between the experiment and a simulation that its source location is inconsistent with this distance. Figure 2.11(c) shows a case where two maxima exist in a correlation image. Although the maxima are located closely, the one with the second highest value coincides with the actual source location.

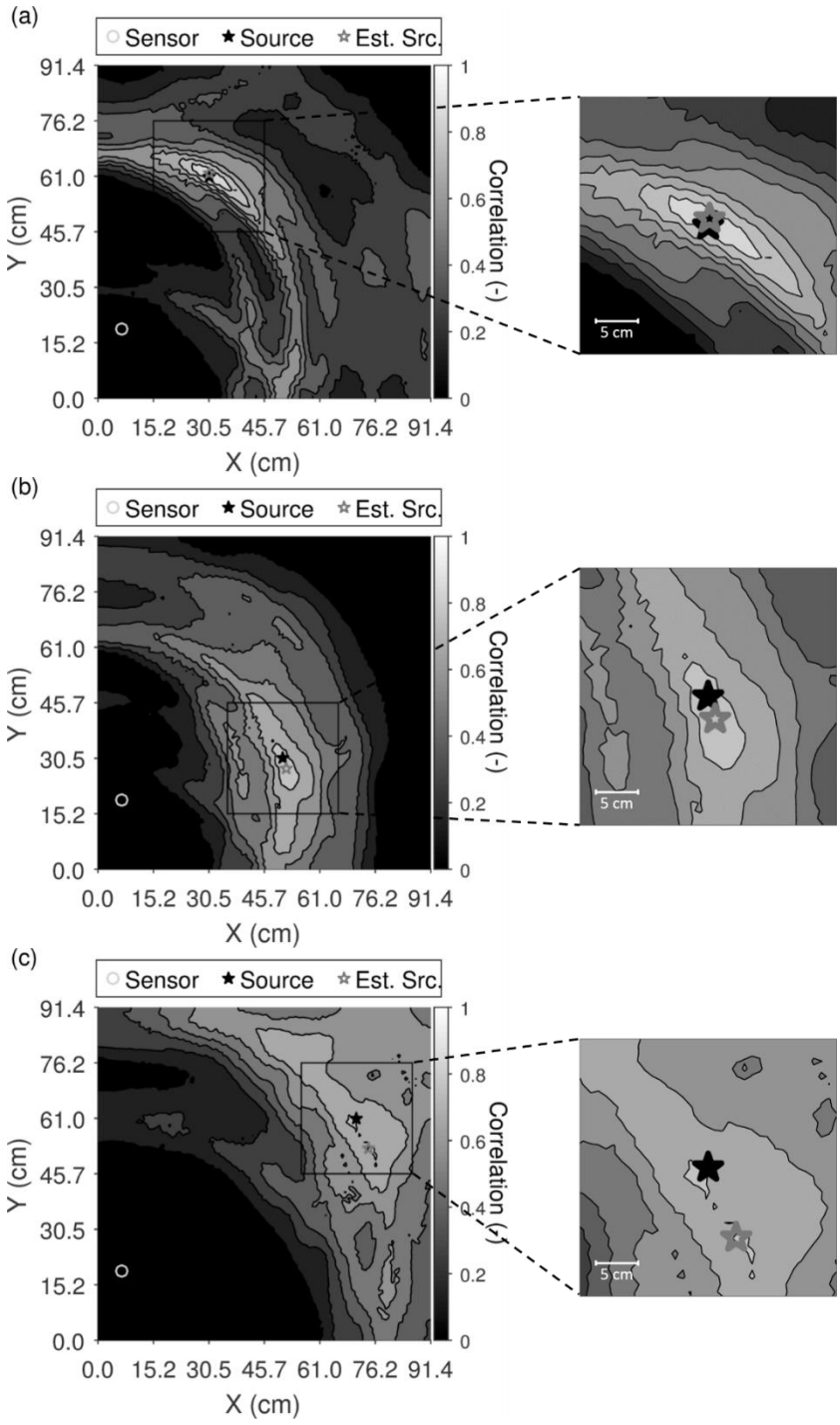

Figure 2.11: Correlation images for PLB tests at coordinate: (a) (30.5 cm, 61.0 cm), (b) (50.8 cm, 30.5 cm), and (c) (71.1 cm, 61.0 cm).

2.5.4. Overall accuracy and error

Figure 2.12 shows the histogram of error for the source-to-sensor distance estimation applied to the 256 PLB tests. These errors are the difference between the actual source-to-sensor distances and the estimated values in the first step (i.e. step I) of the source localization algorithm. The histogram shows less than 0.5 cm error for 99 tests. The maximum error was 3.5 cm in these estimations. In addition, the average of the absolute error was 0.9 cm, and the bias (i.e. the average error) was -0.1 cm. These results validate the source-to-sensor distance estimation step of the algorithm (step I).

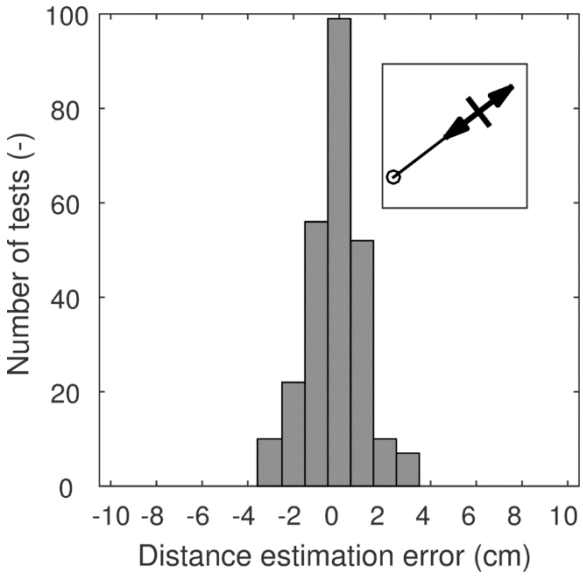

Figure 2.12: Histogram of errors in the source-to-sensor distance estimation of the 256 PLB tests.

Figure 2.13 compares the estimated and actual source locations for the 256 PLB test. The estimated sources with more than 5 cm error are connected to their actual source locations with a line. Overall, all the sources were localized. The maximum localization error was 8.2 cm, and the average error was 2.8 cm. These results show that the proposed algorithm can localize AE sources without any blind zone.

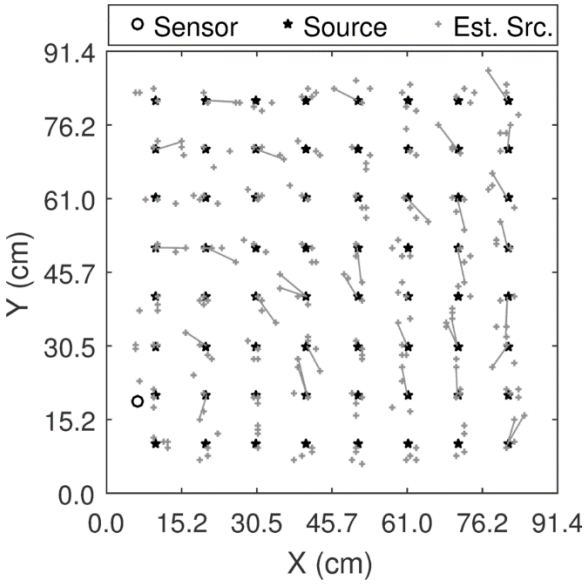

Figure 2.13: Comparison of the actual and estimated source locations; for more than 5 cm error, a line connects the estimated locations to the actual ones.

Figure 2.14(a-b) show the histogram of errors in the radial and tangential directions, respectively. These errors were calculated based on the final localization results (see Figure 2.13). In the radial direction, the maximum error was 3.5 cm and the average of the absolute errors was 1.0 cm. However, in the tangential direction, these numbers were 7.6 and 2.4 cm, respectively. Therefore, as it was observed and explained for the three PLB tests in Figure 2.11, less overall error is expected in the radial direction than the tangential direction.

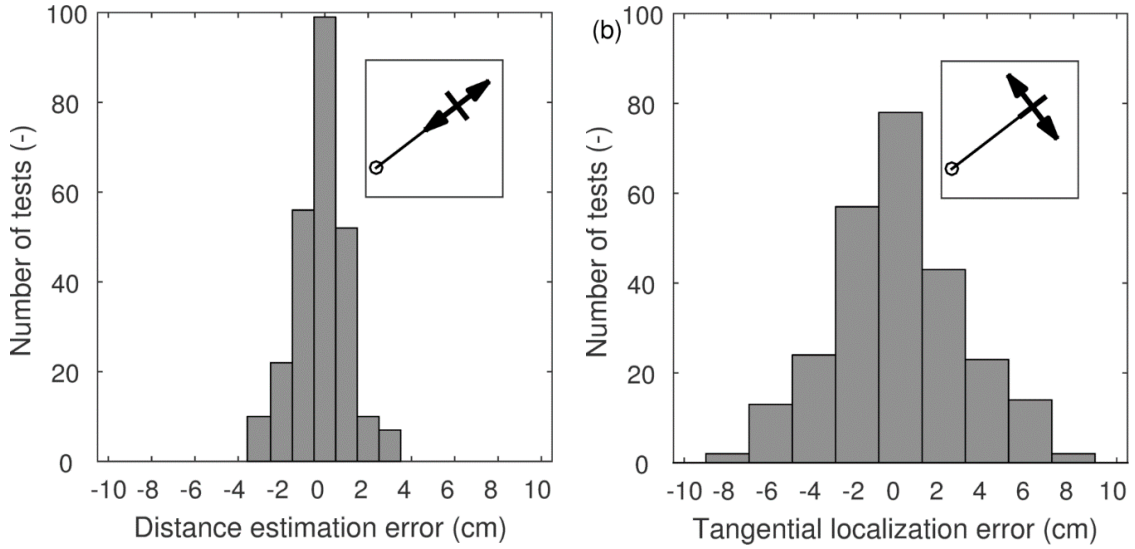

Figure 2.14: Histograms of the two-dimensional localization errors for the 256 PLB tests: (a) radial direction, (b) tangential direction.

Figure 2.15 compares the localization errors of the PLB tests (the test locations are numbered in a row-wise order starting from the lower left test). For the sake of clarity, the four repeated tests that belong to the same test location were sorted based on their localization errors (i.e. the distances between the actual sources and the estimated ones). It could be seen that the localization errors are not the same among the four repeated tests. For example, for test eight, which was performed at (81.3 cm, 10.2 cm), this error varied between 0.7 cm to 6.7 cm. In addition, the errors in most tests (i.e. 241 out of the 256 tests specifically), were more than 0.6 cm. For the 100 × 100-resolution used in the correlation imaging, this error was expected because the distances between the actual sources and their nearest pixels were in the range of 0 cm to 1.3 cm. These results demonstrate that the proposed source localization algorithm is not biased at any specific test point, and thus random localization errors are expected anywhere on the plate.

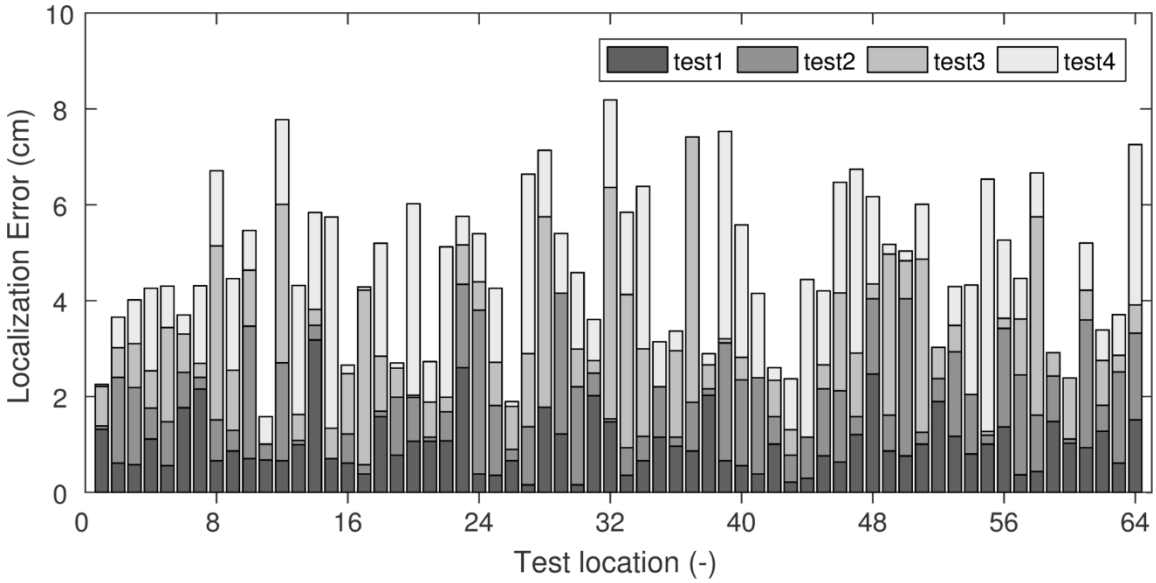

Figure 2.15: Comparison of localization errors; PLB tests were repeated four times at 64 locations.

Table 2.2 studies the effect of the parameter $o_{\max}$ (i.e. the maximum number of reflections that the MP model traces on a propagation path). Two sources were considered: one at the coordinates (30.5 cm, 61.0 cm) and the other one at (50.8 cm, 30.5 cm). These are the same sources shown in Figure 2.11(a-b). However, due to space limitation, fewer details are provided for the second source. The estimated source locations and their corresponding errors demonstrate that a higher $o_{\max}$ slightly improves the final localization results. However, this effect is not significant because the A_0 mode was dominant in the signals, and the earliest time that a second or higher order A_0 arrival appeared in the signals was at $t = 625.7\mu s$, which is outside the range used to calculate the correlations (i.e. $[-200\mu s, 600\mu s]$, see the time range in Figure 2.10). In addition, the table shows that the higher the $o_{\max}$, the higher both the correlations and the computational costs. However, the correlations do not improve after adding the fourth or fifth-order reflections because the earliest S_0 arrivals of these two orders also arrive outside the $[-200\mu s, 600\mu s]$ range.

Moreover, it could be seen in the table that the zero-order MP model cannot localize sources with one sensor. In this case, the model only simulates first arrivals and disregards edge reflections. Therefore, if the parameter $o_{\max}$ is zero, the proposed source localization algorithm can only estimate source-to-sensor distances. In another word, at least two sensors are required to localize sources with a zero-order MP model (similar to the modal acoustic emission algorithms). According to these results, the values of one, two, or three are recommended for $o_{\max}$. These values should be selected based on the tradeoff between their computational cost and accuracy.

Table 2.2: Effect of the parameter $o_{\max}$ on the correlation and localization results.

	(30.5 cm, 61.0 cm) †						(50.8 cm, 30.5 cm) †	
oMax (-)	Earliest S_0 (μs)	Earliest A_0 (μs)	Corre-lation (-)	Computa-tion time‡ (-)	Estimated source (cm)	Error (cm)	Estimated source (cm)	Error (cm)
0	-13.8	50.6	0.687	1	N.A.*	N.A.*	N.A.*	N.A.*
1	183.5	385.8	0.879	4	(31.5, 60.8)	1.38	(52.6, 35.2)	5
2	324.7	625.7	0.901	16	(31.5, 60.8)	1.38	(51.7, 27.9)	2.7
3	525.6	967.2	0.905	57	(30.6, 61.7)	0.73	(51.7, 27.9)	2.7
4	671.9	1215.7	0.905	163	(30.6, 61.7)	0.73	(51.7, 27.9)	2.7
5	872.1	1555.9	0.905	413	(30.6, 61.7)	0.73	(51.7, 27.9)	2.7

† The actual source location

‡ The values are normalized

* Not available

2.5.5. Computation time

The computation time of the proposed source localization algorithm can be broken down into the time spent on the following tasks: (a) the source-to-sensor distance estimation, (b) the MP ray tracking, (c) the MP model and correlation imaging. A MATLAB implementation of the algorithm on a core i5 PC respectively spent 1.5 seconds, 3 minutes, and 3 seconds on average to complete the above-mentioned tasks at 100 × 100

pixel resolution. It needs to be noted that only one run of the MP ray-tracking algorithm is enough for the lifespan of the SHM system. From each pixel, the MP ray-tracking algorithm calculates all possible paths from that pixel to the sensor and stores them in a database. The same database can be reused for all future localizations. Therefore, the actual localization time for each AE event was 1.5+3=4.5 seconds.

2.6. DISCUSSIONS AND CONCLUSIONS

This paper presented a novel, single-sensor AE source localization algorithm for thin metallic plates. The algorithm leverages AE reflections and reverberations as well as the multimodal nature of plate waves. Three key steps were considered. First, a least squares problem was introduced to estimate source-to-sensor distances. Then, an analytical model (the MP model) was proposed to reconstruct the edge-reflected arrivals of AE signals based on their first arrivals. Finally, the correlation analysis between the simulated and experimental signals was used to identify the location of AE sources. Experiments were performed on an aluminum plate to validate the approach, and very good results were achieved. It was observed that the algorithm, unlike many traditional algorithms, has no blind zones and can localize AE sources located even very close to the edges or corners of the plate. This is particularly important because those areas are potentially more prone to fatigue cracks than the rest of the plate. In addition, the accuracy and speed of the proposed approach demonstrated its potential for real-time SHM applications as well as implementation in micro electromechanical systems (MEMS) (Kabir et al. 2015a; b) or wireless SHM systems (Zahedi and Huang 2014).

Despite the promising results presented in this paper, the proposed algorithm has some limitations. First, real-world plate-like structures often consist of several bounded

panels surrounded by stiffer geometric features such as stiffeners and stringers that are not considered in the current MP model. To achieve the overarching goal in the future of this study, which is the monitoring of such bounded panels with only one sensor, future studies should extend the current MP model by developing reflection models for stiffeners and stringers. In addition, the current model is only applicable for thin isotropic plates. Therefore, future studies should include plates that are consists of multiple layers, variable thicknesses, and/or composite materials. Moreover, the uncertainties observed in the localization results needs to be further studied, and thus future studies should take a probabilistic approach to quantify such uncertainties. Finally, the experiments were conducted in a laboratory setting with controlled environmental conditions. Therefore, future studies should extend the model to account for temperature variations. More importantly, on-field experiments need to be conducted to verify the robustness of the approach for real applications.

2.7. ACKNOWLEDGMENTS

This work was supported by the National Science Foundation under the grant number CMMI-1333506.

2.8. ADDITIONAL MATERIALS: FIRST ARRIVAL DETECTION AND WAVE PACKET ISOLATION

The subsequent subsections provide the details of the techniques used to: (1) identify the first S_0 and A_0 arrivals time and (2) isolate the first-arrival packets from the rest of the signal. For the S_0 mode, because it is the faster mode, the Akaike information criterion (AIC) is used (Kurz et al. 2005). Although the AIC is very powerful in identifying the very first wave packet in a signal (in this case S_0), it is not as robust in identifying the

first arrival of the A_0 mode. Therefore, a threshold-based technique is proposed to identify the high-amplitude first arrivals of A_0 that come after the low-amplitude arrivals of S_0.

2.8.1. Akaike information criterion (AIC)

The AIC is a statistical measure, which its global minimum coincides with the first-arrival time of the fastest propagating mode (the S_0 mode in this paper) (Kurz et al. 2005):

$$AIC(t_i) = (t_i)\log(\text{var}(r_{ii})) + (t_N - t_{i+1})\log(\text{var}(r_{iii})) \tag{2.20}$$

where $ii \in [1, i]$, $iii \in [i+1, N]$. The parameter N is the length of the signal **r**. The time at which the AIC is at its global minimum corresponds to the time of the first arrival in the signal **r**. Figure 2.16 shows the values of the AIC for the signal shown in Figure 2.6.

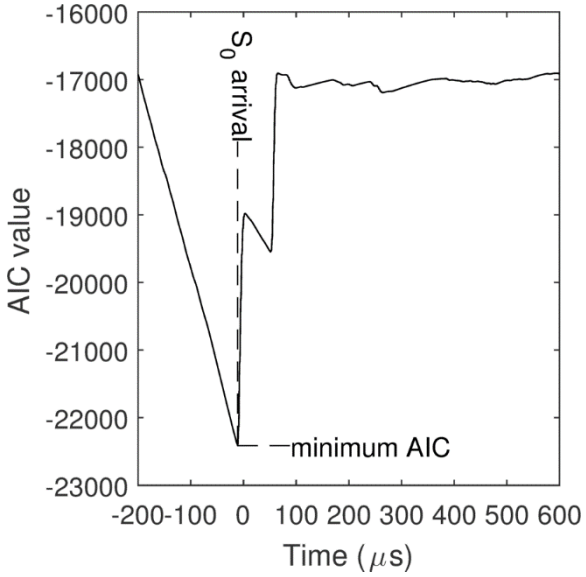

Figure 2.16: Autoregressive AIC; the global minimum of the AIC coincides with the S_0 arrival time.

Once the first-arrival time is identified, the first S_0 wave packet can be isolated. Figure 2.17 visualizes the isolated S_0 packet on the signal shown in Figure 2.6. The time corresponding to the minimum of the AIC is shown as point 2. At point 4, the envelope of

the signal, **e**, reaches to its first local minimum after point 2. The first S_0 wave packet is defined from point 1, which is the first zero crossing before point 2, to point 3, which is the last zero crossing before point 4.

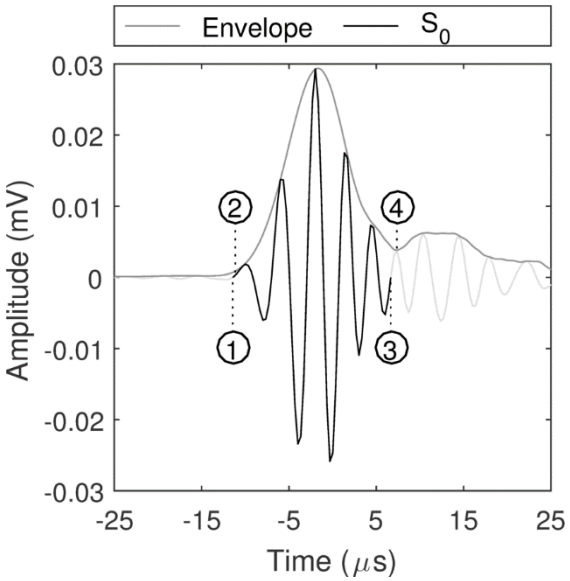

Figure 2.17: The first S_0 wave packet: point 1 is the first zero crossing before point 2; point 2 corresponds to the minimum of AIC; point 3 is the first zero crossing before point 4; point 4 is the local minimum of the signal's envelope.

2.8.2. Threshold-base technique

A customized threshold-base technique is used to identify the arrival time of the first A_0 mode. According to this technique, during the post processing, a secondary threshold is defined relative to the peak amplitude of the signal, which is calculated based on the maximum value of the signal's envelope. This secondary threshold is set at the two-third of the just-defined peak amplitude. Then, a half sine is fitted to the portion of the envelope that ranges from the threshold crossing to the next adjacent peak. Finally, the zero crossing of the fitted half sine is determined by extrapolation. The time of this zero crossing defines the arrival time. Figure 2.18 visualizes the technique on the signal shown in

Figure 2.6. Point 4 is the secondary threshold crossing, and point 3 is the arrival time determined by zero crossing of the extrapolated half sine.

To isolate the first A_0 wave packet, the local minima of the signal's envelope are used. First, the two nearest minima before and after the arrival time are identified (i.e., respectively, point 1 and 6 in Figure 2.18). Then, the isolated wave packet is defined from point 2, which is the first zero crossing of the signal after point 1, to point 5, which is the last zero crossing before point 6.

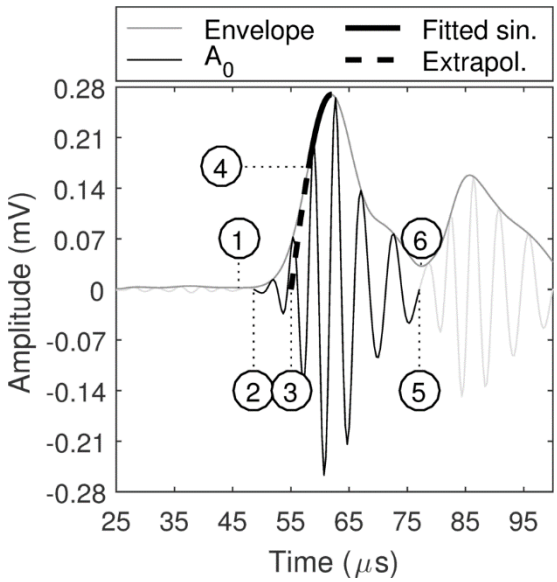

Figure 2.18: The time of arrival and the wave packet of an A_0 arrival: 1) the local minimum of the envelope 2) the first zero crossing after 1; 3) the considered time of arrival 4) the secondary threshold crossing; 5) the last zero crossing before 6; 6) the local minimum of the envelope.

2.9. ADDITIONAL MATERIALS: SAMPLE OUTPUTS OF THE MP RAY-TRACKING ALGORITHM

The detailed outputs of the MP ray-tracking algorithm for the source and receiver shown in Figure 2.8 are presented in Table 2.3.

Table 2.3: Output of the MP ray-tracking algorithm for the source† and receiver‡ shown in Figure 2.8.

Path number (i)	Number of reflections (o_i)	Reflection sequence*	1st reflection coordinates (cm)	2nd reflection coordinates (cm)	3rd reflection coordinates (cm)	Propagation distance d_i (cm)
1	0	[]	-	-	-	48.4
2	1	[L]	(0.0,26.3)	-	-	55.8
3	1	[B]	(12.1, 0.0)	-	-	83.6
4	2	[B, L]	(2.4, 0.0)	(0.0, 5.3)	-	88.1
5	1	[T]	(23.3,91.4)	-	-	105.7
6	2	[T, L]	(19.6,91.4)	(0.0,36.8)	-	109.3
7	2	[T, B]	(25.3,91.4)	(9.6, 0.0)	-	143.0
8	3	[T, L, B]	(22.5,91.4)	(0.0, 5.3)	(1.4, 0.0)	145.7
9	1	[R]	(91.4,43.5)	-	-	151.9
10	2	[R, L]	(91.4,44.9)	(0.0,20.7)	-	164.2
11	2	[R, B]	(91.4,27.6)	(41.1, 0.0)	-	166.5
12	3	[R, B, L]	(91.4,30.2)	(31.4, 0.0)	(0.0,15.8)	177.8
13	2	[T, R]	(73.8,91.4)	(91.4,79.0)	-	178.6
14	3	[T, R, L]	(77.5,91.4)	(91.4,82.4)	(0.0,23.2)	189.2
15	3	[T, R, B]	(62.1,91.4)	(91.4,63.1)	(26.1, 0.0)	203.0
16	2	[L, R]	(0.0,54.8)	(91.4,36.3)	-	211.2
17	3	[L, R, B]	(0.0,49.2)	(91.4,13.8)	(55.6, 0.0)	221.9
18	3	[L, R, L]	(0.0,55.1)	(91.4,37.7)	(0.0,20.3)	223.7
19	2	[B, T]	(23.9, 0.0)	(14.1,91.4)	-	226.1
20	3	[B, T, L]	(20.5, 0.0)	(5.5,91.4)	(0.0,57.8)	227.8
21	3	[L, T, R]	(0.0,76.1)	(30.9,91.4)	(91.4,61.3)	231.2
22	3	[B, T, B]	(24.9, 0.0)	(16.5,91.4)	(8.1, 0.0)	264.0
23	3	[B, R, T]	(70.1, 0.0)	(91.4,32.9)	(53.4,91.4)	268.1
24	3	[T, B, T]	(27.9,91.4)	(20.2, 0.0)	(12.5,91.4)	286.8
25	3	[R, L, R]	(91.4,53.2)	(0.0,41.5)	(91.4,29.9)	331.6

†Source coordinates: (30.5 cm,61.0 cm) ‡Sensor coordinates: (6.4 cm,19.1 cm)

* L: Left boundary B: Bottom boundary R: Right boundary T: Top boundary

PAPER 3

A Probabilistic Framework for Single-sensor Acoustic Emission Source Localization in Thin Metallic Plates

Arvin Ebrahimkhanlou and Salvatore Salamone[1]

3.1. SYNOPSIS

Tracking edge-reflected acoustic emission (AE) waves can allow the localization of their sources. Specifically, in bounded isotropic plate structures, only one sensor may be used to perform these source localizations. The primary goal of this paper is to develop a three-step probabilistic framework to quantify the uncertainties associated with such single-sensor localizations. According to this framework, a probabilistic approach is first used to estimate the direct distances between AE sources and the sensor. Then, an analytical model is used to reconstruct the envelope of edge-reflected AE signals based on the source-to-sensor distance estimations and their first arrivals. Finally, the correlation between the probabilistically reconstructed envelopes and recorded AE signals are used to estimate confidence contours for the location of AE sources. To validate the proposed framework, Hsu-Nielsen pencil lead break (PLB) tests were performed on the surface as

[1]This chapter have been previously published as: Ebrahimkhanlou, Arvin, and Salamone, Salvatore (2017). "A probabilistic framework for single-sensor acoustic emission source localization in thin metallic plates." Smart Materials and Structures, 26(9), 095026, DOI: https://doi.org/10.1088/1361-665X/aa78de
According to the publisher's website: https://publishingsupport.iopscience.iop.org/questions/quick-check-guide-current-author-rights-policy/ accessed on July 25, 2018 inclusion of accepted manuscripts in a research dissertation is permitted. In addition, the dissertation advisor has allowed the direct reproduction of this paper. This is a non-typeset version of the accepted manuscript except for the numbering of sections, figures, tables and equations that have been adapted from the original to satisfy the formatting requirements for a dissertation. The author of this dissertation was the lead investigator conduction the research. The author was responsible of developing theories, designing and conducting experiments, analyzing the results, and writing the journal paper. The coauthors contributed by providing editorial comments.

well as the edges of an aluminum plate. The localization results show that the estimated confidence contours surround the actual source locations. In addition, the performance of the framework was tested in a noisy environment simulated by two dummy transducers and an arbitrary wave generator. The results show that in low-noise environments, the shape and size of the confidence contours depend on the sources and their locations. However, at highly noisy environments, the size of the confidence contours monotonically increases with the noise floor. Such probabilistic results suggest that the proposed probabilistic framework could thus provide more comprehensive information regarding the location of AE sources.

Keywords: modal acoustic emission, uncertainty propagation, probability, source localization, reverberations, plate-like structures, structural health monitoring

3.2. INTRODUCTION

Quantifying the uncertainties of structural-health-monitoring (SHM) techniques is crucial to ensure the safety and performance of structures. This is specifically relevant to the acoustic emission (AE) events of metallic plate structures (e.g., the fatigue cracks and corrosion of ship hulls, bridge girders and gusset plates, or aircraft wings and fuselages (Kundu 2014; Yu et al. 2012; Zárate et al. 2012)), where uncertainties tend to hinder the proper interpretation of localization results. For example, without properly assessing the associated uncertainties, it is nearly impossible to differentiate between multiple individual cracks and a single progressive crack that generates multiple AE events. Such uncertainties are almost never constant. Therefore, it is important to quantify them before making any course of action regarding the conditions of structures.

In order to quantify uncertainties in AE source localization, many studies have proposed multi-sensor, probabilistic algorithms (Dehghan Niri et al. 2014; Dehghan Niri and Salamone 2012; Mallardo et al. 2016; Perelli et al. 2012; Schumacher et al. 2012; Tang et al. 2016; Yan and Tang 2015). For instance, Niri and Salamone (2012) applied the extended Kalman filter to AE signals and probabilistically localized AE sources in a thin metallic plate. In their study, they used and compared three uncertainty propagation techniques. Specifically, they used an analytic approach based on Taylor series approximation, the unscented transform (UT), and Monte Carlo (MC) simulations to quantify the uncertainties in group velocities by propagating the systematic uncertainties of the wavelet transform through dispersion curves. Later, Niri et al. (2014) used unscented and nonlinear Kalman filters to extend their approach to anisotropic composite panels. As another example, Perelli et al. (2012) used the Cramer-Rao bound to quantify the uncertainty in their source localization algorithm that uses edge-reflected reverberations and their dispersive nature. More recently, Yan and Tang (Yan and Tang 2015) developed a Bayesian approach that uses MC to probabilistically localized AE source in thin metallic structures. In another study, Mallardo et al. (2016) also used MC simulations to calculate the Bayesian probability of detecting an impact given the probability of sensor malfunctioning. However, all these studies require multiple AE sensors.

In the literature, two deterministic approaches exist for single-sensor AE source localization. The first approach is to use the edge-reflected reverberations in the tails (codas) of the signals as virtual sensors. For example, Ciampa and Meo (2011) developed a data-driven algorithm that implicitly includes edge reflections in source localization. The second approach is to measure the time of arrival for multiple plate wave modes in AE signals (i.e. Lamb wave modes) and consider them as additional sensors (Jiao et al. 2004;

Surgeon and Wevers 1999). However, this approach, which is called *modal acoustic emission* (MAE), can only estimate source-to-sensor distances and requires an additional sensor to localize sources (Jiao et al. 2008; Toyama et al. 2001). As an alternative approach, in our previous publication, we developed a single sensor source localization algorithm that deterministically combines the two approaches (Ebrahimkhanlou and Salamone 2017d).

Despite these notable contributions, little-to-no research has been done to quantify the uncertainty of single-sensor source localization. In order to fulfill this gap, we propose and develop a probabilistic framework that uses only one sensor. For this purpose, any conventional AE sensor with broadband and omnidirectional characteristics could be used. To achieve such one-sensor localization and quantify localization uncertainties, the framework probabilistically integrates a modal acoustic emission (MAE) approach with a reflection-based (multipath) approach. The idea is to first quantify uncertainties in the MAE approach and then propagate them through the reflection-based approach. In particular, the MAE approach is formulated as a total-least-squares (TLS) regression problem. This approach is based on time-of-arrival measurement on the continuous wavelet transform (CWT) of AE signals. The time and frequency uncertainties in these measurements are propagated to the TLS problem and thus accounted in its solution. In addition, the uncertainties in the TLS solution are quantified using CRB. Such uncertainties are then propagated through the reflection-based approach and their probability distribution is calculated using the kernel density estimation (KDE) (see (Yu and Su 2012)) technique.

The organization of the paper is as follows. Section 3.3 reviews the probabilistic and statistical methods used in the source localization framework. Although this section briefly discusses the physical meaning of such methods, their actual AE application could

be found in section 3.4. Accordingly, section 3.4 develops the framework by implementing the probabilistic and statistical methods in the context of AE source localization. Next, section 3.5 presents the experimental setup used to validate the framework. After that, experimental results are shown and discussed in sections 2.5, 3.6, and 3.7, respectively. Finally, section 3.8 summarizes the contribution of the paper, its limitations, and suggestions for future work.

3.3. PROBABILISTIC AND STATISTICAL METHODS

In this section, we provide an overview of the probabilistic and statistical methods that are used in the third section of the paper. Specifically, we first review the total-least-squares (TLS) regression and the Cramer-Rao bound (CRB). In terms of AE applications, the former estimates the time at which an AE event occurs to ultimately estimate the source-to-sensor distance, and the latter quantifies the uncertainty of such estimation. Then, we present three uncertainty propagation techniques and discuss their differences: the Taylor series approximation, the unscented transform (UT), and Monte Carlo (MC) simulations. In terms of AE applications, such techniques enable the source localization framework to account for the effect of systematic and random uncertainties on localization results and find the most-probable zone, rather a single point, at which an AE source is located. Finally, the kernel density estimation (KDE) technique is reviewed. In terms of AE applications, this technique allows us to fit a probability distribution to the most-probable zone for the source location.

We use the following notation throughout this paper: all scalar quantities are typeset in italic lower-case letters, while vectors and matrices are respectively shown with bold lower-case letters and bold capitalized letters. In this notation, italic lower-case letters with

single or double-figure numerical subscripts indicate the elements of a vector or a matrix, respectively. In addition, a bar '¯' on top of the symbol of a quantity indicates its expected value (i.e. arithmetic mean), while a tide '~' specifies the noisy measurement of that quantity. To show the estimated, noise-free version of a quantity, a hat '^' is used on top of its symbol. Moreover, the standard deviation of a variable is indicated as the Greek letter 'σ' with that variable in subscript.

3.3.1. Total least squares (TLS)

The TLS is a regression technique that accounts for uncertainties in both the dependent and independent variables of the regression. Consider an over-determinate system of equations with *n* equations and one unknown:

$$\boldsymbol{\delta} = \mathbf{c}\ \tau_{\mathrm{e}} \tag{3.1}$$

where $\mathbf{c}$ and $\boldsymbol{\delta}$ are known $n \times 1$ vectors, and τ_{e} is an unknown scalar. The TLS returns the noise-free estimates of $\mathbf{c}$ and $\boldsymbol{\delta}$ (i.e. $\hat{\mathbf{c}}$ and $\hat{\boldsymbol{\delta}}$) as well as an estimate of the unknown τ_{e} (i.e. $\hat{\tau}_{\mathrm{e}}$).

In terms of the AE application for plate-like structures, each waveform consists of multiple frequencies and Lamb wave modes that each propagates at a different group velocity, and arrives to an AE sensor at different time. If we perform n measurements of the group velocities and time of arrivals at different frequencies and for different modes, the vector $\mathbf{c}$ contains the mean subtracted group velocity measurements, and the vector $\boldsymbol{\delta}$ contains the mean subtracted, element-wise product of the group velocity and time of arrival measurements. Since $\boldsymbol{\delta}$ is in the unit of length, the regression parameter τ_{e} is expected to be in the unit of time. Later, in section 3.4, we will show that the physical meaning of this parameter is the time at which an AE event occurs.

Generally, each measurement is a noisy observation of actual values. In this case, if we assume an additive noise, the noisy observations of $\mathbf{c}$ and $\boldsymbol{\delta}$ have the following form:

$$\begin{cases} \tilde{\mathbf{c}} = \mathbf{c} + \eta_1 \mathbf{1} \\ \tilde{\boldsymbol{\delta}} = \boldsymbol{\delta} + \eta_2 \mathbf{1} \end{cases} \tag{3.2}$$

If we further assume a zero-mean, bivariate Gaussian noise, $\boldsymbol{\eta} = [\eta_1,\ \eta_2]^T$:

$$\boldsymbol{\eta} = \mathrm{N}(\mathbf{0},\ \boldsymbol{\Sigma}) \tag{3.3}$$

In this equation, $\boldsymbol{\Sigma}$ is the covariance matrix of the noise, and N stands for the normal (Gaussian) distribution. Later, in section 3.4, we will further explain that the underlying reason for assuming zero-mean, additive Gaussian noise is because of the type of the complex mother wavelet that is used for the measurements.

The TLS requires a diagonal $\boldsymbol{\Sigma}$ unless a transformation, $\mathbf{T}_\Sigma$, rotates and scales $\tilde{\mathbf{c}}$ and $\tilde{\boldsymbol{\delta}}$ on the principal components of $\boldsymbol{\Sigma}$. To find $\mathbf{T}_\Sigma$, the singular value decomposition (SVD) of $\boldsymbol{\Sigma} = \mathbf{U}_\Sigma \mathbf{S}_\Sigma \mathbf{U}_\Sigma^{\mathrm{T}}$ is used:

$$\mathbf{T}_\Sigma = \mathbf{S}_\Sigma^{-\frac{1}{2}} \mathbf{U}_\Sigma \tag{3.4}$$

We define matrix $\tilde{\mathbf{A}} = [\tilde{\mathbf{c}}, \tilde{\boldsymbol{\delta}}]$ and from hereafter refer to it as the design matrix. Accordingly, the transformed design matrix, $\tilde{\mathbf{A}}_\Sigma$, is calculated as:

$$\tilde{\mathbf{A}}_\Sigma = \tilde{\mathbf{A}} \mathbf{T}_\Sigma^{\mathrm{T}} \tag{3.5}$$

The transform $\mathbf{T}_\Sigma$ is the identity matrix when the noise in $\tilde{\mathbf{c}}$ is independent of the noise in $\tilde{\boldsymbol{\delta}}$ (i.e. $\boldsymbol{\Sigma}$ is diagonal). Therefore, no rotation or scaling is applied.

To find the noise-free estimates of $\mathbf{c}$ and $\boldsymbol{\delta}$ (i.e. $\hat{\mathbf{c}}$ and $\hat{\boldsymbol{\delta}}$), we use the SVD of the transformed design matrix, $\tilde{\mathbf{A}}_\Sigma = \mathbf{U}_\mathrm{A} \mathbf{S}_\mathrm{A} \mathbf{V}_\mathrm{A}^{\mathrm{T}}$. Matrix $\mathbf{S}_\mathrm{A}$ is an $n \times 2$ diagonal matrix, which its diagonal elements are in a descending order. The TLS estimates the noise-free design matrix in the transformed coordinates as (Golub and van Loan 1980; Markovsky and Van Huffel 2007):

$$\hat{\mathbf{A}}_{\Sigma} = \mathbf{U}_{\mathrm{A}} \mathrm{diag}(s_{11}, 0) \mathbf{V}_{\mathrm{A}}^{\mathrm{T}} \tag{3.6}$$

In this equation, s_{11} is the first element in the first column of $\mathbf{S}_{\mathrm{A}}$. Therefore, the estimated noise-free design matrix in the original coordinates is represented as:

$$\hat{\mathbf{A}} = \hat{\mathbf{A}}_{\Sigma} \mathbf{T}_{\Sigma}^{-\mathrm{T}} \tag{3.7}$$

where $\hat{\mathbf{A}} = [\hat{\mathbf{c}}, \hat{\boldsymbol{\delta}}]$. In another words, $\hat{\mathbf{c}}$ and $\hat{\boldsymbol{\delta}}$ are the first and second columns of the matrix $\hat{\mathbf{A}}$, respectively.

To find $\hat{\tau}_{\mathrm{e}}$, vector $\mathbf{v}_{\Sigma}$ is defined to be the last column of $\mathbf{V}_{\mathrm{A}}$. In addition, we define vector $\mathbf{w}_{\Sigma}$ in a way that $\mathbf{v}_{\Sigma}$ and $\mathbf{w}_{\Sigma}$ be perpendicular to each other (Golub and van Loan 1980; Markovsky and Van Huffel 2007):

$$\mathbf{w}_{\Sigma} \perp \mathbf{v}_{\Sigma} \tag{3.8}$$

In the original coordinates:

$$\mathbf{w} = \mathbf{T}_{\Sigma}^{-1} \mathbf{w}_{\Sigma} \tag{3.9}$$

Finally, the TLS estimates τ_{e} as:

$$\hat{\tau}_{\mathrm{e}} = \frac{w_2}{w_1} \tag{3.10}$$

where w_1 and w_2 are the first and second elements of the vector $\mathbf{w}$, respectively.

3.3.2. Cramer-Rao bound (CRB)

The CRB is an uncertainty-quantification technique that estimates a lower bound on the uncertainty of an unbiased estimator (in this case the TLS). In terms of the AE application, the covariance of $\hat{\tau}_{\mathrm{e}}$ is at least equal or greater than the inverse of its Fisher information matrix (Cramér 1946; Rao 1945):

$$\mathrm{cov}(\hat{\tau}_{\mathrm{e}}) \geq \mathbf{F}(\tau_{\mathrm{e}})^{-1} \tag{3.11}$$

where $\mathbf{F}(\tau_{\mathrm{e}})$ is the Fisher information matrix. To define the Fisher information matrix in terms of $\mathbf{c}$ and $\boldsymbol{\delta}$ measurements (i.e. $\tilde{\mathbf{c}}$ and $\tilde{\boldsymbol{\delta}}$) and their unbiased estimates (i.e. $\hat{\mathbf{c}}$ and $\hat{\boldsymbol{\delta}}$),

we first use Eq. (3.1) and Eq. (3.2) to define the *i*-th measurement noise, $\boldsymbol{\eta}^{(i)}$, in terms of τ_{e}:

$$\begin{cases} \eta_1^{(i)} = \tilde{c}_i - \hat{c}_i \\ \eta_2^{(i)} = \tilde{\delta}_i - \hat{\delta}_i = \tilde{\delta}_i - \hat{c}_i \tau_{\mathrm{e}} \end{cases} \tag{3.12}$$

Then, we can defined the Fisher information matrix as:

$$\mathbf{F}(\tau_{\mathrm{e}}) = \mathrm{E}((\frac{\partial}{\partial \tau_{\mathrm{e}}} \log(f(\boldsymbol{\eta}^{(i)} : \tau_{\mathrm{e}})))(\frac{\partial}{\partial \tau_{\mathrm{e}}} \log(f(\boldsymbol{\eta}^{(i)} : \tau_{\mathrm{e}})))^{\mathrm{T}}) \tag{3.13}$$

where $\mathrm{E}(\cdot)$ is the expected value (i.e. arithmetic mean). The log likelihood of the additive bivariate Gaussian noise, $\boldsymbol{\eta} = \mathrm{N}(\mathbf{0}, \boldsymbol{\Sigma})$, can be calculated as:

$$\log f(\boldsymbol{\eta}^{(i)}) = -\frac{1}{2}(\boldsymbol{\eta}^{(i)})^{\mathrm{T}} \boldsymbol{\Sigma}^{-1} (\boldsymbol{\eta}^{(i)}) - \frac{1}{2}\log(2\pi|\boldsymbol{\Sigma}|) \tag{3.14}$$

Substituting Eq. (3.12) into Eq. (3.14):

$$\log f(\boldsymbol{\eta}^{(i)}) = -\frac{1}{2}[\tilde{c}_i - \hat{c}_i \ , \ \tilde{\delta}_i - \hat{c}_i \tau_{\mathrm{e}}] \boldsymbol{\Sigma}^{-1} [\tilde{c}_i - \hat{c}_i \ , \ \tilde{\delta}_i - \hat{c}_i \tau_{\mathrm{e}}]^{\mathrm{T}} - \frac{1}{2}\log(2\pi|\boldsymbol{\Sigma}|) \tag{3.15}$$

By differentiating with respect to τ_{e}:

$$\frac{\partial}{\partial t_{\mathrm{e}}} \log f(\boldsymbol{\eta}^{(i)} : \tau_{\mathrm{e}}) = [0 \ , \ \hat{c}_i] \boldsymbol{\Sigma}^{-1} [\tilde{c}_i - \hat{c}_i \ , \ \tilde{\delta}_i - \hat{c}_i \tau_{\mathrm{e}}]^{\mathrm{T}} \tag{3.16}$$

Therefore, substituting Eq. (3.16) into Eq. (3.13) yields the following:

$$\mathbf{F}(\tau_{\mathrm{e}}) = \frac{1}{n} \sum_{i=1}^{n} \left([0 \ , \ \hat{c}_i] \boldsymbol{\Sigma}^{-1} [\tilde{c}_i - \hat{c}_i \ , \ \tilde{\delta}_i - \hat{c}_i \tau_{\mathrm{e}}]^{\mathrm{T}} \right)^2 \tag{3.17}$$

Since usually τ_{e} is unknown, one can use its unbiased estimator, $\hat{\tau}_{\mathrm{e}}$ to approximate the Fisher information matrix. Therefore, Eq. (3.17) and Eq. (3.11) yield a lower bound on the variance of $\hat{\tau}_{\mathrm{e}}$:

$$\sigma_{\hat{\tau}_{\mathrm{e}}}^2 \geq \mathbf{F}(\hat{\tau}_{\mathrm{e}})^{-1} \tag{3.18}$$

3.3.3. Uncertainty propagation

Uncertainty propagation techniques are statistical approaches that calculate the effect of input uncertainties on the output of a function. In terms of the AE application of

this paper, one example for such a function is the function that takes $\hat{\tau}_e$ and returns the propagation distance (we will formally introduce this function in section 3.4). Another function is our reflection-based localization model that takes the propagation distance and returns the location of the source. To lay the foundation for such applications, this section reviews three uncertainty propagation techniques: the Taylor series approximation, the unscented transform (UT), and Monte Carlo (MC) simulations. In order to formally present these three techniques in a uniform mathematical notation, consider an arbitrary vector function $\mathbf{g}(\mathbf{z})$, which takes the random variable $\mathbf{z}$ as its input. The following subsections show how each uncertainty propagation technique uses the statistics of $\mathbf{z}$ (i.e. mean, $\bar{\mathbf{z}}$; covariance, $\boldsymbol{\Sigma}_z$; etc.) to estimate uncertainties in the output of the function $\mathbf{g}(\mathbf{z})$.

3.3.3.1. Taylor series approximation

The Taylor series approximation of $\mathbf{g}(\mathbf{z})$ around $\bar{\mathbf{z}}$ estimates the statistics of $\mathbf{g}(\mathbf{z})$ if the partial derivatives of $\mathbf{g}(\mathbf{z})$ are available (Julier and Uhlmann 2004). Only the first order derivatives, $\mathcal{D}\mathbf{g}(\mathbf{z})$, are used to estimate the mean of $\mathbf{g}(\mathbf{z})$ when the higher order partial derivatives of $\mathbf{g}(\mathbf{z})$ are either zero or negligible:

$$\begin{aligned} \mathrm{E}(\mathbf{g}(\mathbf{z})) &= \mathrm{E}(\mathbf{g}(\bar{\mathbf{z}}) + \mathcal{D}\mathbf{g}(\mathbf{z})|_{\mathbf{z}=\bar{\mathbf{z}}}\,(\mathbf{z}-\bar{\mathbf{z}})) \\ &= \mathbf{g}(\bar{\mathbf{z}}) + \mathcal{D}\mathbf{g}(\mathbf{z})|_{\mathbf{z}=\bar{\mathbf{z}}}\,(\mathrm{E}(\mathbf{z})-\bar{\mathbf{z}}) \\ &= \mathbf{g}(\bar{\mathbf{z}}) \end{aligned} \tag{3.19}$$

where the operator $\mathcal{D}$ is the Jacobian of $\mathbf{g}(\mathbf{z})$ (i.e. $\mathcal{D}_{ij}\mathbf{g}(\mathbf{z}) = \partial g_i / \partial z_j$). Similarly, the covariance of $\mathbf{g}(\mathbf{z})$ is estimated as:

$$\begin{aligned} \mathrm{cov}(\mathbf{g}(\mathbf{z})) &= \mathrm{E}((\mathbf{g}(\mathbf{z}) - \mathrm{E}(\mathbf{g}(\mathbf{z})))(\mathbf{g}(\mathbf{z}) - \mathrm{E}(\mathbf{g}(\mathbf{z})))^{\mathrm{T}}) \\ &= \mathrm{E}((\mathbf{g}(\bar{\mathbf{z}}) + \mathcal{D}\mathbf{g}(\mathbf{z})|_{\mathbf{z}=\bar{\mathbf{z}}}\,(\mathbf{z}-\bar{\mathbf{z}}) - \mathbf{g}(\bar{\mathbf{z}}))(\mathcal{D}\mathbf{g}(\mathbf{z})|_{\mathbf{z}=\bar{\mathbf{z}}}\,(\mathbf{z}-\bar{\mathbf{z}}))^{\mathrm{T}}) \\ &= \mathcal{D}\mathbf{g}(\mathbf{z})|_{\mathbf{z}=\bar{\mathbf{z}}}\,\mathrm{E}((\mathbf{z}-\bar{\mathbf{z}})(\mathbf{z}-\bar{\mathbf{z}})^{\mathrm{T}})(\mathcal{D}\mathbf{g}(\mathbf{z})|_{\mathbf{z}=\bar{\mathbf{z}}})^{\mathrm{T}} \\ &= \mathcal{D}\mathbf{g}(\mathbf{z})|_{\mathbf{z}=\bar{\mathbf{z}}}\,\boldsymbol{\Sigma}_z(\mathcal{D}\mathbf{g}(\mathbf{z})|_{\mathbf{z}=\bar{\mathbf{z}}})^{\mathrm{T}} \end{aligned} \tag{3.20}$$

In terms of the AE application, the following is the physical meaning of Eq. (3.20) if applied on a function, which takes $\hat{\tau}_e$ and returns the propagation distance (the first example in section 3.3.3): the standard deviation of the propagation distance is the product of $\sigma_{\hat{\tau}_e}$ and the derivative of the function at $\hat{\tau}_e$. Later in section 3.4, we will calculate this derivative.

Although this technique provides a fast, closed-form approximation for linear functions, it is not applicable to all functions. Alternative approaches should be considered under the following conditions: 1) the partial derivatives of the function are intractable to calculate or 2) the higher order partial derivatives are non-negligible.

3.3.3.2. Unscented transform (UT)

Unscented transform is a fast uncertainty propagation technique for slightly nonlinear functions. Considering the arbitrary function $\mathbf{g}(\mathbf{z})$ again, this technique is applicable when $\mathbf{z}$ has a Gaussian distribution, and $\mathbf{g}(\mathbf{z})$ is smooth and nonsingular. In terms of AE applications, examples of such functions include dispersion curves, where uncertain frequencies causes uncertain group velocities (Dehghan Niri and Salamone 2012). According to this technique, deterministic samples, named *sigma points*, are selected from the distribution of $\mathbf{z}$. The number of the sigma points is twice the dimension of $\mathbf{z}$. The vectors of sigma points are represented as (see (Julier and Uhlmann 2004)):

$$\begin{aligned} \mathbf{z}_{\mathrm{UT}}^{(2i-1)} &= \overline{\mathbf{z}} - \sqrt{N_z}(\mathbf{S}_z^{\frac{1}{2}}\mathbf{U}_z^{\mathrm{T}})_i \\ \mathbf{z}_{\mathrm{UT}}^{(2i)} &= \overline{\mathbf{z}} + \sqrt{N_z}(\mathbf{S}_z^{\frac{1}{2}}\mathbf{U}_z^{\mathrm{T}})_i \end{aligned} \tag{3.21}$$

where $\mathbf{S}_z$ and $\mathbf{U}_z$ are the SVD of $\boldsymbol{\Sigma}_z = \mathbf{U}_z\mathbf{S}_z\mathbf{U}_z^{\mathrm{T}}$, the notation $(\cdot)_i$ indicates the i-th column of the matrix $\mathbf{S}_z^{\frac{1}{2}}\mathbf{U}_z^{\mathrm{T}}$, and N_z is the dimension of $\mathbf{z}$. These sigma points are propagated through the function $\mathbf{g}$: $\mathbf{g}_{\mathrm{UT}}^{(i)} = \mathbf{g}(\mathbf{z}_{\mathrm{UT}}^{(i)})$. The mean of the propagated sigma points estimates

the mean of $\mathbf{g}(\mathbf{z})$, $\mathrm{E}(\mathbf{g}(\mathbf{z})) = \mathrm{E}(\mathbf{g}_{\mathrm{UT}})$. Similarly, covariance of the propagated sigma points estimates the covariance of $\mathbf{g}(\mathbf{z})$, $\mathrm{cov}(\mathbf{g}(\mathbf{z})) = \mathrm{cov}(\mathbf{g}_{\mathrm{UT}})$.

3.3.3.3. Monte Carlo (MC) simulations

Monte Carlo simulations are a family of uncertainty propagation techniques that are generally applicable to all functions and input distributions. However, they usually require more computation than the two previously discussed techniques. Therefore, MC simulations are used when other uncertainty propagation techniques are not applicable. Such cases include highly nonlinear functions and/or sophisticated input distributions.

Let $\mathbf{g}(\mathbf{z})$ satisfy either or both of the just mentioned conditions, and $\mathbf{z}_{\mathrm{MC}}^{(i)}$ be a random sample (an *MC sample*) from the distribution of $\mathbf{z}$. The corresponding propagated point (an *MC point*) thus is $\mathbf{g}_{\mathrm{MC}}^{(i)} = \mathbf{g}(\mathbf{z}_{\mathrm{MC}}^{(i)})$. Similar to the UT, the statistics of $\mathbf{g}(\mathbf{z})$ are estimated from the statistics of the MC points. In addition, since MC does not impose any assumptions on the distribution of $\mathbf{g}(\mathbf{z})$, MC points estimate the actual distribution of $\mathbf{g}(\mathbf{z})$.

In terms of the AE application, this paper applies MC simulations on a reflection-based source localization approach. Since this source localizer is a function of source-to-sensor distances, each MC sample physically means the distance between the source and the sensor, while each MC point physically means a pair of coordinates for the source location.

3.3.3.4. Kernel density estimation (KDE)

This technique is a non-parametric method to estimate a probability distribution from samples of a distribution. In terms of the AE application, the KDE could be used to estimate a probability distribution for the location of AE sources. Assume that a N_{KDE} number of samples are available from a probability density function (pdf) $p(\mathbf{z})$, where

$\mathbf{z}_{\mathrm{KDE}}^{(i)}$ is one of these samples. In this AE application, $\mathbf{z}_{\mathrm{KDE}}^{(i)}$ is an MC point that estimates the location of the source, and $p(\mathbf{z})$ is the desired probability distribution for the location of AE sources. The KDE estimate of this pdf is defined as:

$$\hat{p}(\mathbf{z}) = \frac{1}{N_{\mathrm{KDE}}} \sum_{i=1}^{N_{\mathrm{KDE}}} \varphi_{\mathrm{KDE}}(\mathbf{z} - \mathbf{z}_{\mathrm{KDE}}^{(i)}) \tag{3.22}$$

where φ_{KDE} is a Gaussian kernel function(Duda et al. 2012):

$$\varphi_{\mathrm{KDE}}(\boldsymbol{\xi}) = \frac{1}{2\pi\sqrt{|\mathbf{R}|}} \exp(-\frac{\boldsymbol{\xi}^{\mathrm{T}}\mathbf{R}^{-1}\boldsymbol{\xi}}{2}) \tag{3.23}$$

The covariance matrix $\mathbf{R}$ is defined as $\mathbf{R} = b_{KDE}\mathbf{I}$, where $\mathbf{I}$ is the identity matrix and b_{KDE} is an adjustable parameter named bandwidth.

3.4. PROBABILISTIC SOURCE LOCALIZATION FRAMEWORK

In this section, we propose a novel probabilistic source-localization framework, which works with as few as only one sensor. Figure 3.1 illustrates the flowchart of this three-step framework. In the first step, the direct source-to-sensor distance is estimated, and the uncertainties of this estimation are quantified. The estimated value is then used in the second step to reconstruct the edge-reflected wave packets in the coda (tail) of an AE signal. Finally, the reconstructed packets and quantified uncertainties are used in the third step to probabilistically localize AE sources. The subsequent subsections discuss each step in details.

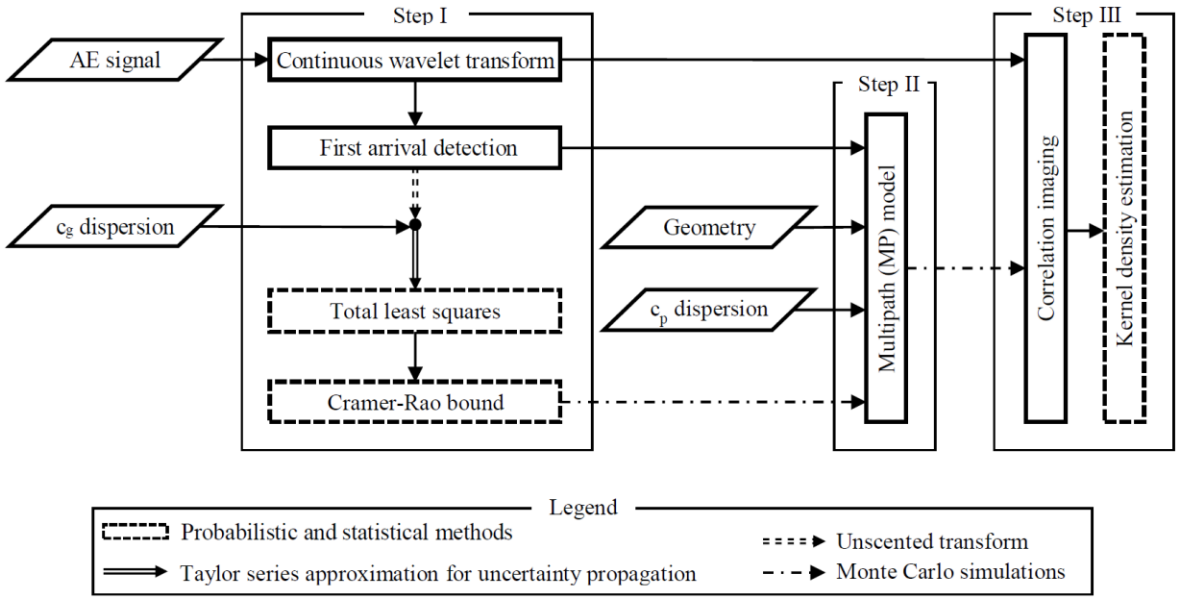

Figure 3.1: Flowchart of the proposed probabilistic source localization framework; the dash lines indicate probabilistic and statistical modules.

3.4.1. Probabilistic estimation of the source-to-sensor distance (step I)

The first step in the proposed framework is to estimate direct source-to-sensor distances and quantify the uncertainty of this estimation. Figure 3.2 visualizes a direct source-to-sensor distance, which hereafter is shown with parameter *d*. To estimate *d*, we first calculate the continuous wavelet transform (CWT) of input AE signals. Then, the first arrival time of the two fundamental Lamb wave modes (S_0 and A_0) is measured at different frequencies. After that, the dispersion curves of the plate are used to form a system of equations based on the group velocities and the time of first arrivals. The idea is to use the frequency-dependent differences in the group velocities and thus in the time of flights (TOF) of the Lamb waves that travel the distance d with different group velocities. To solve this system of equations, the TLS method is used. Finally, the CRB method is used to estimate a lower bound on the uncertainties in the estimation of *d*. The details of such calculations are discussed in the following subsections.

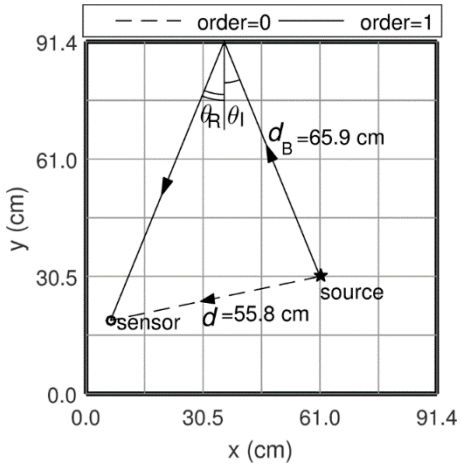

Figure 3.2: Schematic of a direct source-to-sensor path (zero-order) and an indirect path (first-order); the angles of incident and reflection as well as a source-to-edge distance are indicated on the indirect path.

3.4.1.1. Continuous wavelet transform (CWT)

The CWT has been widely used in various SHM applications (Mostavi et al. 2017; Sarrafi and Mao 2016), including acoustic emission source localization (Dehghan Niri and Salamone 2012; Ebrahimkhanlou and Salamone 2017b; Hyunjo Jeong and Young-Su Jang 2000). For an input signal $s(t')$, the CWT is defined as:

$$C_W(f,t) = \frac{1}{\sqrt{s_W(f)}} \int_{-\infty}^{+\infty} s(t') \Psi_W^*\left(\frac{t'-t}{s_W(f)}\right) dt' \tag{3.24}$$

In this equation, $s_W(f)$ is the dimensionless scale parameter, t is the translation parameter, and Ψ_W^* is the complex conjugate of the mother wavelet $\Psi_W(t)$ (Mallat 1998). The complex Morlet mother wavelet is defined as:

$$\Psi_W(t) = \frac{1}{\sqrt{\pi f_b}} \exp\left(2\pi f_c jt - \frac{\tau^2}{f_b}\right) \tag{3.25}$$

where f_b and f_c are non-dimensional parameters for bandwidth frequency and the central frequency, respectively. The dimensionless scale parameter of CWT, $s_W(f)$, is defined as:

$$s_{\mathrm{W}}(f) = \frac{f_c \cdot f_{\mathrm{s}}}{f} \tag{3.26}$$

In this equation, f_{s} is the sampling frequency of $s(t')$.

Systematic uncertainties are involved in the time and frequency resolution of the CWT. According to the Heisenberg inequality:

$$\sigma_t^2 \sigma_f^2 \geq \frac{1}{16\pi^2} \tag{3.27}$$

where σ_t^2 and σ_f^2 are respectively the variances in time and frequency resolutions. These uncertainties for the complex Morlet CWT are respectively defined as (Tse and Lai 2007):

$$\sigma_t = \frac{f_{\mathrm{c}}\sqrt{f_{\mathrm{b}}}}{2f} \tag{3.28}$$

and

$$\sigma_f = \frac{f}{2\pi f_{\mathrm{c}}\sqrt{f_{\mathrm{b}}}} \tag{3.29}$$

If we calculate the CWT coefficients for the frequencies stored in a frequency vector $\tilde{\mathbf{f}}$, due to these systematic uncertainties, $\tilde{\mathbf{f}}$ will be the noisy version of the actual frequencies (i.e. the vector **f**). To evaluate such uncertainties, we use Eq. (3.29) and show them with vector $\boldsymbol{\sigma}_f$. In addition, the real part of the CWT output is an unbiased Gaussian, band-pass filter with central frequency and standard deviation of f and σ_f, respectively (Tse and Lai 2007):

$$r(f,t) = \mathrm{Re}(C_{\mathrm{W}}(f,t)) \tag{3.30}$$

Moreover, the modules ($|\cdot|$) of the CWT coefficients is the envelope of the filtered signal:

$$e(f,t) = |C_{\mathrm{W}}(f,t)| \tag{3.31}$$

3.4.1.2. First arrival detection

The arrival time of the fastest propagating mode can be measured using the Akaike information criterion (AIC). The AIC is an autoregressive, which its minimum coincides with the arrival time of the fastest mode. For a vector of digitized signal **r**, the AIC is defined as (Kurz et al. 2005; Sedlak et al. 2013):

$$AIC(t_i) = (t_i)\log(\text{var}(r_{1,\dots,i})) + (t_{N_r} - t_{i+1})\log(\text{var}(r_{i+1,\dots,N_r})) \tag{3.32}$$

where N_r is the overall number of data points in **r** (i.e. the length of the vector **r**), t_i is the time corresponding to the *i*-th data point in the signal, and var() is the variance.

In plate-like structures, the fastest propagating mode at frequencies below the first cut-off frequency is the first symmetric Lamb wave mode (S_0). The first arrival of this mode propagates directly from the source to the sensor without any reflection. Therefore, the AIC can be used to effectively measure the arrival time of the first S_0 packet. . This measurement is performed on the real-value part of the CWT coefficients, $r(\tilde{f}_i, t)$, at the frequency $\tilde{f}_i$. However, this technique is not applicable to A_0 arrivals. To measure the arrival time of the first A_0 in $r(\tilde{f}_i, t)$, we define a threshold-based technique, which uses the higher amplitude of A_0 relative to S_0. The detailed implementation of the threshold-based technique is presented in section 3.10. In addition, this section presents the process of isolating the first arrival packets from the rest of the signal.

3.4.1.3. Total least squares formulation

Different frequencies and modes travel the source-to-sensor distance d at different group velocities. If an AE event occurs at the time τ_e (Ebrahimkhanlou and Salamone 2017d):

$$\text{diag}(\mathbf{c}_g)(\boldsymbol{\tau} - \tau_e\mathbf{1}) = d\mathbf{1} \tag{3.33}$$

where $\mathbf{c}_g$ is the vector of group velocities, $\boldsymbol{\tau}$ is the vector of arrival time, $\mathbf{1}$ is a vector with all elements equal to 1 (i.e. $(\boldsymbol{\tau} - \tau_e \mathbf{1})$ is the TOF vector). The elements of the vectors $\mathbf{c}_g$ and $\boldsymbol{\tau}$ correspond to the frequencies listed in $\tilde{\mathbf{f}}$. To calculate $\mathbf{c}_g$, the dispersion curves of the plate are used. In addition, the first arrival detection techniques are used to calculate the vector $\boldsymbol{\tau}$ (see section 3.4.1.2). The both vectors $\mathbf{c}_g$ and $\boldsymbol{\tau}$ may include more than one Lamb wave modes. In such a case, the corresponding vectors of each mode are concatenated to construct $\mathbf{c}_g$ and $\boldsymbol{\tau}$.

To simplify Eq. (3.33), we define the following:

$$\boldsymbol{\delta}_g = \mathrm{diag}(\mathbf{c}_g)\boldsymbol{\tau} \tag{3.34}$$

Therefore, Eq. (3.33) can be rewritten as:

$$\boldsymbol{\delta}_g - \tau_e \mathbf{c}_g = d\mathbf{1} \tag{3.35}$$

The expected value of the two sides of Eq. (3.35) is:

$$\bar{\delta}_g - \tau_e \bar{c}_g = d \tag{3.36}$$

This can be substituted in the right-hand side of Eq. (3.35). Therefore, after rearranging:

$$\boldsymbol{\delta}_g - \bar{\delta}_g \mathbf{1} = (\mathbf{c}_g - \bar{c}_g \mathbf{1})\tau_e \tag{3.37}$$

Finally, to convert Eq.(3.37) to the TLS problem, as stated in the Eq.(3.1), we define:

$$\begin{cases} \mathbf{c} = \mathbf{c}_g - \bar{c}_g \mathbf{1} \\ \boldsymbol{\delta} = \boldsymbol{\delta}_g - \bar{\delta}_g \mathbf{1} \end{cases} \tag{3.38}$$

Therefore, the TLS method can be used to estimate τ_e. However, the covariance of the TLS noise, i.e. the matrix $\boldsymbol{\Sigma}$, first needs to be calculated (see Eq. (3.3)):

$$\boldsymbol{\Sigma} = \begin{bmatrix} \bar{\sigma}_c^2 & \bar{\sigma}_{c\delta}^2 \\ \bar{\sigma}_{c\delta}^2 & \bar{\sigma}_{\delta}^2 \end{bmatrix} \tag{3.39}$$

To calculate $\boldsymbol{\Sigma}$, we first define a noise matrix $\boldsymbol{\Sigma}_i$ for each c_i and δ_i measurements, where the index *i* corresponds to the elements of the vectors $\mathbf{c}$ and $\boldsymbol{\delta}$. In other words, $\boldsymbol{\Sigma}$ is the expected value of n matrices $\boldsymbol{\Sigma}_i$ (i.e. $\boldsymbol{\Sigma} = \mathrm{E}(\boldsymbol{\Sigma}_i)$):

$$\boldsymbol{\Sigma}_i = \begin{bmatrix} \sigma^2_{c_i} & \sigma^2_{c_i\delta_i} \\ \sigma^2_{c_i\delta_i} & \sigma^2_{\delta_i} \end{bmatrix} \tag{3.40}$$

To calculate $\boldsymbol{\Sigma}_\mathrm{i}$, we use the Taylor series approach (see section 3.3.3.1). In this case, uncertainties are propagated through a two-dimensional function defined as $\mathbf{g}_\mathrm{T}(\mathbf{z}) = [c_i \quad \delta_i]^\mathrm{T}$ that satisfies the following:

$$\boldsymbol{\Sigma}_i = \mathrm{cov}(\mathbf{g}_\mathrm{T}(\mathbf{z})) \tag{3.41}$$

We define the input to $\mathbf{g}(\mathbf{z})$ as $\mathbf{z} = [c_i \quad \tau_i]^\mathrm{T}$. The mean of $\mathbf{z}$ is $\bar{\mathbf{z}} = [c_i \quad \tau_i]^\mathrm{T}$ because we assume the wavelet-based measurements of group velocity and first arrivals are unbiased. In addition, its covariance is $\boldsymbol{\Sigma}_\mathrm{z} = [\sigma^2_{c_i} \quad 0; \quad 0 \quad \sigma^2_{\tau_i}]$, where $\sigma^2_{c_i\tau_i}$ is zero because uncertainties in both the arrival time τ_i and the group velocity c_i are independent. In this definition, σ_{c_i} (in vector notation: $\boldsymbol{\sigma}_c$) is equal to $\boldsymbol{\sigma}_{c_g}$ because $\mathbf{c}$ is the summation of a constant (i.e. $-\bar{c}_\mathrm{g}$) with $\mathbf{c}_\mathrm{g}$ (see Eq. (3.38)). To calculate $\boldsymbol{\sigma}_{c_g}$, we use the UT and propagate the frequency uncertainties $\boldsymbol{\sigma}_f$ through dispersion curves (see section 3.3.3.2). To calculate σ_{τ_i}, we use the standard deviation of the CWT resolution in time (i.e. σ_t) at the corresponding frequency to the index (see Eq. (3.28)).

The partial derivatives of $\mathbf{g}(\mathbf{z})$ with respect to $\mathbf{z}$ are defined as:

$$\mathcal{D}\mathbf{g}(\mathbf{z}) = \begin{bmatrix} 1 & 0 \\ \tau_i & c_i \end{bmatrix} \tag{3.42}$$

Using Eq. (3.20) and Eq. (3.41), the covariance $\boldsymbol{\Sigma}_i$ is represented as:

$$\boldsymbol{\Sigma}_i = \begin{bmatrix} \sigma^2_{c_i} & \tau_i\sigma^2_{c_i} \\ \tau_i\sigma^2_{c_i} & \tau_i^2\sigma^2_{c_i} + c_i^2\sigma^2_{\tau_i} \end{bmatrix} \tag{3.43}$$

Therefore, $\sigma^2_{c_i\delta_i} = \tau_i\sigma^2_{c_i}$ and $\sigma^2_{\delta_i} = \tau_i^2\sigma^2_{c_i} + c_i^2\sigma^2_{\tau_i}$. This concludes the calculation of TLS inputs. Thus one can use Eq. (3.10) to calculate $\hat{\tau}_{\mathrm{e}}$.

At this point, the underlying reason for assuming a zero-mean, additive Gaussian noise in Eq. (3.3) can be explained. Since the time and frequency resolution of the complex Morlet mother wavelet are perturbed with a zero-mean, additive Gaussian noise, the uncertainty of $\tilde{\boldsymbol{\tau}}$ also has a similar type of distribution. In addition, we assume the same type of distribution for the uncertainty of $\tilde{\mathbf{c}}_{\mathrm{g}}$ and thus $\tilde{\mathbf{c}}$. This is because of the fact that the group velocity dispersion curves of the fundamental modes are smooth functions of $\tilde{\mathbf{f}}$ with slight nonlinearities. Therefore, $\boldsymbol{\eta}$ is the multiplication of two zero-mean Gaussian distributions and thus has the same type of distribution as well.

Finally, the value of d can be estimated by substituting the $\hat{\tau}_{\mathrm{e}}$ in Eq. (3.36):

$$\hat{d} = \bar{\delta}_{\mathrm{g}} - \hat{\tau}_{\mathrm{e}}\bar{c}_{\mathrm{g}} \tag{3.44}$$

In addition, the TLS provides unbiased estimates of $\hat{\mathbf{c}}$ and $\hat{\boldsymbol{\delta}}$ (see Eq. (3.7)). These estimates can be used to calculate the unbiased estimates of the first arrival time $\hat{\boldsymbol{\tau}}$. To perform such calculations, we first rewrite Eq. (3.34) as $\boldsymbol{\tau} = \mathrm{diag}^{-1}(\boldsymbol{\delta}_{\mathrm{g}})\mathbf{c}_{\mathrm{g}}$, and then use Eq. (3.38) to substitute $\mathbf{c}_{\mathrm{g}}$ and $\boldsymbol{\delta}_{\mathrm{g}}$:

$$\hat{\boldsymbol{\tau}} = \mathrm{diag}^{-1}(\hat{\boldsymbol{\delta}} + \bar{\delta}_{\mathrm{g}}\mathbf{1})(\hat{\mathbf{c}} + \bar{c}_{\mathrm{g}}\mathbf{1}) \tag{3.45}$$

3.4.1.4. Cramer-Rao bound on the estimated values

The CRB can be used to calculate a lower-bound on the uncertainties of both $\hat{\tau}_{\mathrm{e}}$ and $\hat{d}$. To quantify the uncertainties in the estimated time of AE event, one can use Eq. (3.18). However, an additional step needs to be taken before quantifying the uncertainties

in the estimated source-to-sensor distance. Since Eq. (3.44) defines $\hat{d}$ as a linear function of $\hat{\tau}_e$, Eq. (3.20) can be applied to propagate the uncertainties from $\hat{\tau}_e$ to $\hat{d}$:

$$\sigma_{\hat{d}} = \bar{c}_g \sigma_{\hat{\tau}_e} \tag{3.46}$$

Eq. (3.46) states that the uncertainty of source-to-sensor distance estimation is $\bar{c}_g$ times larger than the uncertainty of time of event estimation.

3.4.2. Multipath (MP) model (step II)

The MP model is an analytical model to simulate the envelope of edge-reflected late arrivals. This model was initially developed to simulate the Lamb waves excited by an actuator and scattered by damage (Ebrahimkhanlou et al. 2016b). Recently, we modify this model to reconstruct the envelopes of AE signals based on the geometry of the plate, sensor, and the estimated source-to-sensor distance, d (Ebrahimkhanlou and Salamone 2017d). According to this model, first arrivals and phase-velocity dispersion curves are used to reconstruct late arrivals. The following subsections discuss the four modules of this model: ray tracking, wave propagation, edge reflection, and envelope reconstruction.

3.4.2.1. Ray tracking

In order to track the propagation paths of acoustic waves from their sources to a sensor, we use an algorithm named MP ray tracking algorithm (Ebrahimkhanlou et al. 2016b). The inputs to this algorithm are the following: a) the dimensions of the plate, b) the coordinates of the sensor, and c) a guess for the location of the source (i.e. a two-element vector **x**). The algorithm returns q paths that the number of reflections occurred on them is less than or equal to the specified maximum o_{max}. For each path, the algorithm returns the travel distance, d_i, and the number of reflections, o_i. In such calculations, it is assumed that no mode conversion occurs during the edge reflections. This assumption is

based on the contribution of Gunawan and Hirose (2007). They demonstrated that no mode conversion occurs between symmetric and antisymmetric modes during reflections from the square-cut edges of a plate. In addition, we filtered the high-frequency content to keep the frequencies below the first cut-off frequency of the higher-order Lamb modes. In such a case, the only propagating Lamb wave modes are the two fundamental modes (S_0 and A_0). Therefore, these two modes do not convert to each other. Moreover, according to the Snell's law, this also requires the angles of incident θ_I and reflection θ_R be equal (see Figure 3.2) (Gunawan and Hirose 2007). More information on the MP ray tracking algorithm, such as its flowchart and pseudo code can be found in a previous publication of the authors (Ebrahimkhanlou et al. 2016b).

3.4.2.2. Wave propagation

Let $u_0(t)$ be the out-of-plane displacement of an excited Lamb wave mode at an AE source, then at a distance d_i from the source the out-of-plane displacement can be calculated as (Wilcox 1998):

$$u(d_i,t) = \mathcal{F}^{-1}\{\mathcal{F}\{u_0(t)\}E(k,\omega)H_0^{(1)}(kd_i)\} \tag{3.47}$$

In this equation, $\mathcal{F}\{\cdot\}$ and $\mathcal{F}^{-1}\{\cdot\}$ are the Fourier transform and its inverse, respectively. In addition, $E(k,\omega)$ is the excitability function of the Lamb wave mode for the wavenumber k and angular frequency ω, $H_0^{(1)}(\cdot)$ is the zero-order Hankel function of the first kind, and d_i is the propagation distance of the i-th arrival (see section 3.4.2.1).

To rearrange this equation in terms of a first arrival $u(d,t)$ (see section 3.4.1.2) instead of the out-of-plane displacement at the source, one can calculate the value of $\mathcal{F}\{u_0(t)\}E(k,\omega)$ based on $u(d,t)$ in the left-hand side of Eq. (3.47):

$$\mathcal{F}\{u_0(t)\}E(k,\omega) = \frac{\mathcal{F}\{u(d_i,t)\}}{H_0^{(1)}(kd_i)} \tag{3.48}$$

Evaluating Eq. (3.48) at $d_i = d$ and substituting the calculated value again in Eq. (3.47):

$$u(d_i,t) = \mathcal{F}^{-1}\{\mathcal{F}\{u(d,t)\}\frac{H_0^{(1)}(kd_i)}{H_0^{(1)}(kd)}\} \tag{3.49}$$

Using the approximation of the Hankel function for not-close-to-zero arguments, $H_0^{(1)}(kd_i) = \sqrt{2/(\pi kd_i)}\exp(\mathrm{j}kd_i - \mathrm{j}\pi/4)$, and $k = 2\pi f / c_\mathrm{p}(f)$, Eq. (3.49) simplifies as (Abramowitz and Stegun 1965):

$$u(d_i,t) = \mathcal{F}^{-1}\{(\frac{d_i}{d})^{-0.5}\mathcal{F}\{u(d,t)\}\exp(\frac{2\pi\mathrm{j}f(d_i-d)}{c_\mathrm{p}(f)})\} \tag{3.50}$$

3.4.2.3. Edge reflections

The MP model accounts for the attenuation due to edge reflections by reducing the amplitude of the reflected wave with a coefficient γ (Ebrahimkhanlou et al. 2016a; b). In addition, reflections cause phase-shifts between incident waves $u_\mathrm{I}(d_\mathrm{B},t)$ and reflected waves $u_\mathrm{R}(d_\mathrm{B},t)$:

$$u_\mathrm{R}(d_\mathrm{B},t) = (\gamma)\mathcal{F}^{-1}\{\mathcal{F}\{u_\mathrm{I}(d_\mathrm{B},t)\}\exp(\mathrm{j}\varphi)\} \tag{3.51}$$

In this equation, d_B is the distance from the source to the reflecting boundary (see Figure 3.2), and φ is the phase-shift. Therefore, Eq. (3.50) and (3.51) can be combined to calculate the late arrivals of each mode:

$$u(d_i,t) = \gamma^{o_i}\mathcal{F}^{-1}\{(\frac{d_i}{d})^{-0.5}\mathcal{F}\{u(d,t)\}\exp(\frac{2\pi\mathrm{j}f(d_i-d)}{c_\mathrm{p}(f)} + \mathrm{j}\varphi')\} \tag{3.52}$$

In this equation, φ' is the overall phase-shift due to all reflections that occur on a propagation path.

3.4.2.4. Envelope reconstruction

The filtered AE signals with the CWT are the sum of several AE arrivals, $u(d_i,t)$. Therefore, Eq. (3.52) can be used to reconstruct late arrivals based on their first arrivals. To minimize the effect of the unknown φ', we use the square root of the sum of squares (SRSS) technique instead of the ordinary summation and reconstruct the envelopes of the filtered AE signals:

$$\mathbf{e}(\mathbf{x}) = \underset{i=1}{\overset{q}{\mathrm{SRSS}}}(u_{\mathrm{S}_0}(d_i,t),\ u_{\mathrm{A}_0}(d_i,t)) \tag{3.53}$$

In this equation, $u_{\mathrm{S}_0}(d_i,t)$ and $u_{\mathrm{A}_0}(d_i,t)$ are the envelopes of the i-th S_0 and A_0 arrivals, respectively (see section 3.4.1.2). In addition, the MP ray tracking algorithm determines q and d_i (see section 3.4.2.1).

3.4.3. Probabilistic localization (Step III)

In this last step, correlation imaging is applied on the reconstructed envelopes to localize AE sources. Such localization is repeated for the Monte-Carlo (MC) simulations used to propagate uncertainties through the MP model. Then, the kernel density estimation (KDE) technique is used to estimate the probability density function (pdf) of the estimated source locations.

3.4.3.1. Correlation imaging

Correlation imaging is a technique to identify the location of AE sources (Ebrahimkhanlou et al. 2016b; Park et al. 2012; Quaegebeur et al. 2011). In this technique, a uniformly spaced grid of points is considered. These points are the pixels of the correlation image and the guesses for source locations (see section 3.4.2.1). For each pixel, a reconstructed signal $\mathbf{e}(\mathbf{x})$ is tested against an experimental signal $\mathbf{e}$, and their correlation, $\rho(\mathbf{x})$, is assigned to the pixel value:

$$\rho(\mathbf{x}) = \frac{\sum_{i=1}^{n_s} (e_i(\mathbf{x}) - \bar{e}(\mathbf{x}))(e_i - \bar{e})}{\sqrt{\sum_{ii=1}^{n_s} (e_{ii}(\mathbf{x}) - \bar{e}(\mathbf{x}))^2 \sum_{iii=1}^{n_s} (e_{iii} - \bar{e})^2}} \tag{3.54}$$

In this equation, $\mathbf{e}$ is the envelope of the filtered signal with the CWT at a specific frequency (i.e. $e(f,t)$). The same frequency is also used to reconstruct $\mathbf{e}(\mathbf{x})$. In addition, n_s is the length of the vectors $\mathbf{e}(\mathbf{x})$ and $\mathbf{e}$. Finally, the pixel with the highest correlation is the estimated source location.

3.4.3.2. Confidence contour estimation

The proposed confidence contour estimation technique quantifies the uncertainties that the MC simulations propagate. The MC simulations carry these uncertainties from the source-to-sensor distance estimation and propagate them through the MP model and correlation imaging. In each MC simulation, a sample is taken from the distribution of the estimated source-to-sensor distance. Then, an estimated source location $\mathbf{x}_i$ is calculated based on this sample. According to this technique, the distribution of *m* number of such MC points represents the uncertainty in the estimated source location. To quantify this uncertainty and estimate its pdf, the KDE technique (see 3.3.3.4) is used:

$$p(\mathbf{x}) = \frac{1}{m} \sum_{i=1}^{m} \varphi(\mathbf{x} - \mathbf{x}_i) \tag{3.55}$$

To calculate a confidence contour, C_α, for a given confidence level α, the pdf is intersected with a horizontal plane. The height of the plane is selected iteratively until the confined volume between the plane and the pdf satisfies this condition:

$$\oint_{C_\alpha} p(\mathbf{x})\, \mathrm{d}\mathcal{A} = \alpha \tag{3.56}$$

The confidence contour is the intersection line of the plane and the pdf.

3.5. EXPERIMENTAL SETUP

To validate the developed source localization framework, experiments were conducted on a 91.4 cm × 91.4 cm × 0.318 cm 6061-T6 aluminum plate. To simulate acoustic emission sources, Hsu-Nielsen (1977) pencil lead break (PLB) tests were performed. In each PLB test, a 3-mm-protruded, 2H lead of a 0.5 mm mechanical pencil was broken at a 45-degree angle with respect to the plate. In particular, two sets of experiments were performed: a) in a noise-free environment and b) in a noisy environment.

In the noise-free experiments, a total of 96 PLB tests were performed on the surface and edges of the plate. Specifically, 64 on-the-surface tests were conducted at the nodes of an 8x8 uniformly spaced grid, and 32 on-the-edge tests were performed at the intersections of the grid with the edges of the plate (see Figure 3.3). In the case of the on-the-edge tests, the pencil lead was placed approximately at the middle of the thickness of the plate.

In the experiments with added noise, one of the on-the-surface pencil lead break tests ($x_1 = 61.0$ cm, $x_2 = 30.5$ cm) was repeated at different noise floors. Specifically, six noise floors were simulated between 60 dB to 70 dB at the steps of 2 dB. To simulate the noisy environment, two dummy transducers with central frequencies of 60 kHz and 150 kHz (Physical Acoustics R6α and R15α) were used, respectively. The R6α was attached to the plate at the coordinates $x_1 = 45.7$ cm, $x_2 = 12.7$ cm, and the R15α transducer was attached at the coordinates $x_1 = 45.7$ cm, $x_2 = 78.7$ cm. In addition, a two-channel arbitrary wave generator (Agilent) was used to excite the dummy transducers with white noise. To simulate different noise floors, the excitation amplitude of the dummy transducers was adjusted in such a way that results in the desired noise floors in the acquired AE signals.

To detect the AE signals, a broadband sensor (Physical Acoustics PICO) was attached to the plate at the coordinates $x_1 = 6.4$ cm and $x_2 = 19.1$ cm. The signals were

amplified 40dB before being digitized by an eight-channel data-acquisition system (Mistras Micro Express). The sampling frequency was 2 MHz, and the signals were filtered before the digitization with a 20 kHz high-pass and a 400 kHz low-pass analog filters. Finally, the AE signals were post-processed in MATLAB, and the Rayleigh-Lamb equations were numerically solved to calculate the group velocity and phase velocity dispersion curves of the plate according to (Rose 2004).

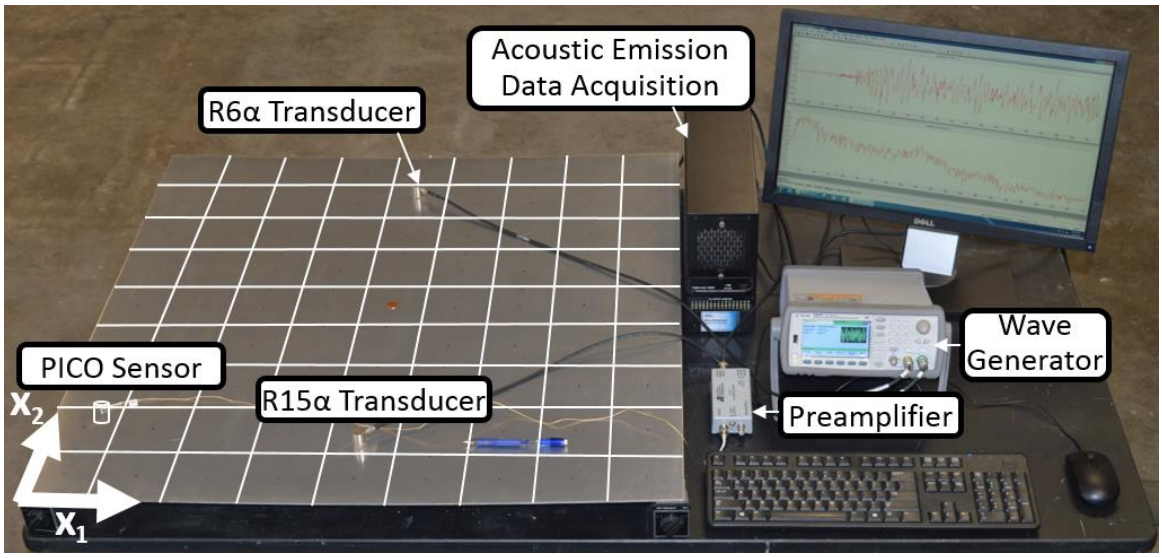

Figure 3.3: Experimental setup; the grid indicates the location of the PLB tests.

3.6. RESULTS

This section presents experimental results for the pencil lead break (PLB) tests performed to simulate AE sources. Specifically, results are presented for noise-free and noisy environments. To separate the effect of environmental noise and systematic uncertainties, the majority of the presented results belong to the noise-free case. In addition, the overall performance and computational cost of the proposed framework are discussed at the end of this section.

For the noise-free case, we first present the intermediate and final results of the probabilistic source-to-sensor distance estimation (step I). Then, we show results for the analytical reconstruction of late arrivals on one of the AE signals and compared it to the experimental signal (step II). We finally visualize the process of calculating confidence contours on the Monte Carlo (MC) simulations (step III). To provide intermediate results throughout this section, the AE signal of the PLB test at the coordinates $x_1 = 61.0$ cm and $x_2 = 30.5$ cm was used.

3.6.1. Probabilistic estimation of source-to-sensor distances

Figure 3.4 visualizes the output of the continuous wavelet transform (CWT) for the AE signal generated at $x_1 = 61.0$ cm and $x_2 = 30.5$ cm. Specifically, Figure 3.4(a) shows $e(f,t)$ and Figure 3.4(b) shows $r(f,t)$ at 100, 175, 250, and 325 kHz. Both figures visualize the dispersive nature of the signal as well as the higher amplitude of the A_0 mode. Figure 3.4(b) also shows the first arrival time measurements, the estimated time of AE event $\hat{\tau}_e$, and the time of the trigger. As it can be seen in the figure, $\hat{\tau}_e$ is always a negative number because it is defined relative to the trigger time. To calculate the CWT coefficients, we selected a frequency vector $\tilde{\mathbf{f}}$ on the sensitive frequency band of the AE sensor. This vector contains 400 equally spaced frequencies from 25 kHz to 400 kHz. In addition, $f_b = 0.5$ and $f_c = 5$ were used for the bandwidth frequency and the central frequency parameters of the CWT, respectively.

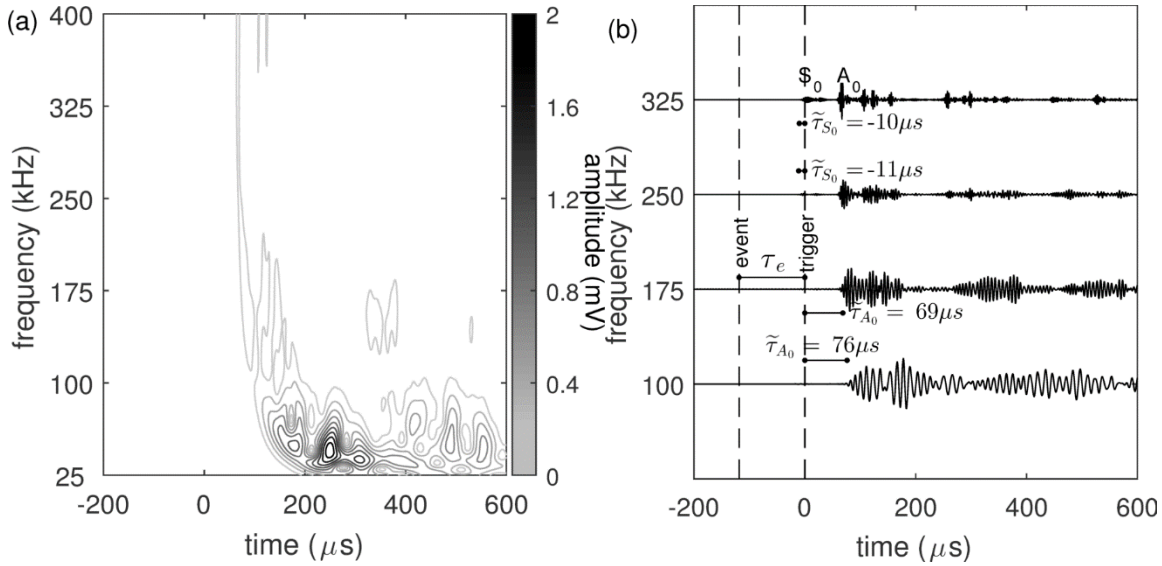

Figure 3.4: The CWT of the AE signal generated at $x_1 = 61.0$ cm and $x_2 = 30.5$ cm: (a) the modules of the CWT and (b) the real part of the CWT.

Figure 3.5 visualizes the Heisenberg uncertainties in the time and frequency resolutions of the complex Morlet CWT used in this study ($f_b = 0.5$ and $f_c = 5$). Eq. (3.28) and (3.29) are used to calculate these uncertainties. As it can be seen in the figure, the uncertainty in the frequency resolution increases at higher frequencies, whereas the uncertainty in the resolution of time decreases.

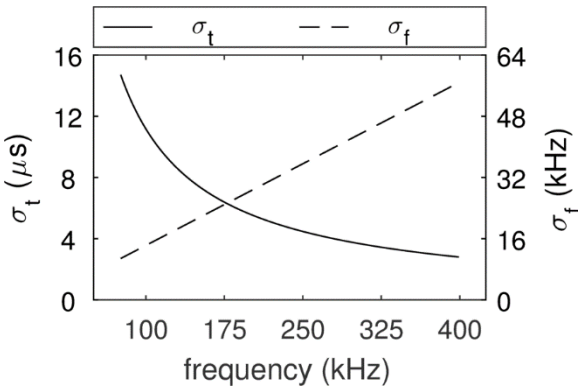

Figure 3.5: Uncertainty in time and frequency resolution of the CWT ($f_b = 0.5$ and $f_c = 5$).

Figure 3.6 shows the measured first arrival time $\tilde{\boldsymbol{\tau}}$ and its uncertainty bound $\boldsymbol{\sigma}_\tau$ at the frequencies listed in the vector $\tilde{\mathbf{f}}$. To measure $\tilde{\boldsymbol{\tau}}$, the first arrival time detection techniques were used. (see 3.4.1.2 and section 3.10). For the S_0 mode, such measurements (i.e. the vector $\tilde{\boldsymbol{\tau}}_{S_0}$) were performed at the frequencies equal to or above 250 kHz because this mode had a negligible amplitude at other lower frequencies (see Figure 3.4(b)). However, the first arrival time of the A_0 mode $\tilde{\boldsymbol{\tau}}_{A_0}$ was measured at the frequencies equal to or below 250 kHz because the higher amplitude of S_0 reflections at higher frequencies might affect the threshold-based technique used for the A_0 mode. To construct $\tilde{\boldsymbol{\tau}}$, the vectors $\tilde{\boldsymbol{\tau}}_{S_0}$ and $\tilde{\boldsymbol{\tau}}_{A_0}$ were concatenated: $\tilde{\boldsymbol{\tau}} = [\tilde{\boldsymbol{\tau}}_{S_0}^{\mathrm{T}}, \tilde{\boldsymbol{\tau}}_{A_0}^{\mathrm{T}}]^{\mathrm{T}}$. In quantifying the uncertainties, it was assumed that the uncertainty in the time of arrivals is due to the Heisenberg uncertainties in the resolution of time. Therefore, Eq. (3.28) was used to quantify the arrival time uncertainties. Similar to the construction of $\tilde{\boldsymbol{\tau}}$, the uncertainties of the two modes were concatenated to construct the vector $\boldsymbol{\sigma}_\tau = [\boldsymbol{\sigma}_{\tau_{S_0}}^{\mathrm{T}}, \boldsymbol{\sigma}_{\tau_{A_0}}^{\mathrm{T}}]^{\mathrm{T}}$. Since an attempt to measure the time of arrivals may fail at some frequencies (due to noise and other reasons explained in section 3.10), the unsuccessful measurements are eliminated from all the measurement vectors (e.g., $\tilde{\boldsymbol{\tau}}$) as well as the uncertainty vectors (e.g., $\boldsymbol{\sigma}_\tau$).

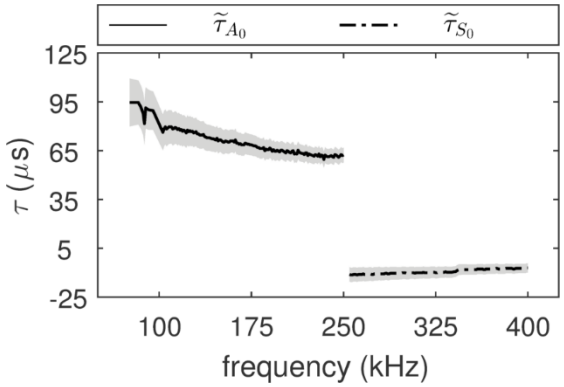

Figure 3.6: First arrival time measurement $\tilde{\boldsymbol{\tau}}$; the shaded areas represent the standard deviations ($\boldsymbol{\sigma}_\tau$).

Figure 3.7 visualized the process of uncertainty propagation through the dispersion curves of the plate. For this purpose, we used the unscented transform (UT) (see section 3.3.3.2) and sampled three sigma points from the probability distribution of the frequency uncertainties $\mathrm{N}(f,\sigma_f)$ (see section 3.4.1.1), where N stands for the normal distribution. These sigma points were propagated through the dispersion curves, and the mean and the standard deviation of their corresponding group velocities were calculated. Although the process is demonstrated for the frequency of 250 kHz, a similar process was repeated for the other frequencies listed in the vector $\tilde{\mathbf{f}}$.

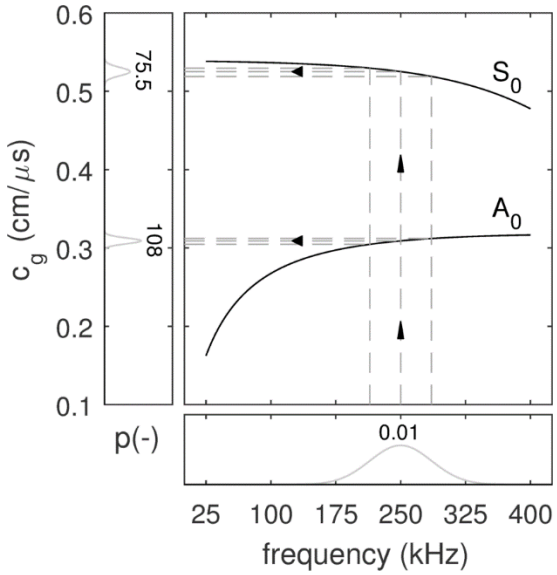

Figure 3.7: Uncertainty propagation through dispersion curves at the frequency of 250 kHz.

Figure 3.8 shows the group velocities $\tilde{\mathbf{c}}_g$ and their uncertainty bounds $\boldsymbol{\sigma}_c$ at the frequencies listed in the vector $\tilde{\mathbf{f}}$. To calculate $\tilde{\mathbf{c}}_g$ and $\boldsymbol{\sigma}_c$, the visualized process in Figure 3.7 was repeated for all of the frequencies listed in $\tilde{\mathbf{f}}$ and the calculated means and standard deviations were stored in separate vectors. To be consistent with the vector $\tilde{\boldsymbol{\tau}}$, only the results corresponding to the frequencies equal to or above 250 kHz were stored for the S_0 mode. For the A_0 mode also, only the results corresponding to the frequencies

equal to or below 250 kHz were stored. Finally, the group velocity vectors of the two modes were concatenated: $\tilde{\mathbf{c}}_{\mathrm{g}} = [\tilde{\mathbf{c}}_{\mathrm{g}_{S_0}}^{\mathrm{T}}, \tilde{\mathbf{c}}_{\mathrm{g}_{A_0}}^{\mathrm{T}}]^{\mathrm{T}}$. Similarly, the uncertainty vector of the group velocities was constructed from the standard deviations: $\boldsymbol{\sigma}_{\mathrm{c}} = [\boldsymbol{\sigma}_{\mathrm{c}_{S_0}}^{\mathrm{T}}, \boldsymbol{\sigma}_{\mathrm{c}_{A_0}}^{\mathrm{T}}]^{\mathrm{T}}$.

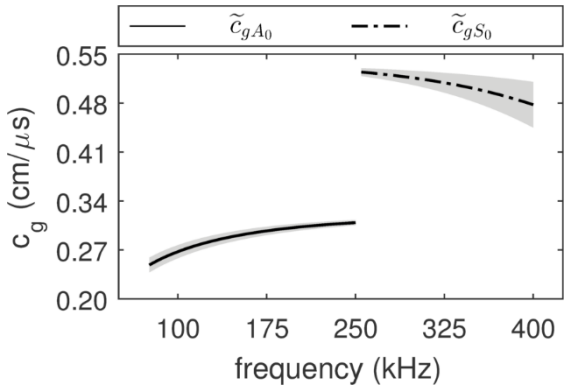

Figure 3.8: S_0 and A_0 group velocities; the shaded areas represent the standard deviations ($\boldsymbol{\sigma}_c$).

Figure 3.9 shows the values of $\tilde{\tilde{\boldsymbol{\delta}}}_{\mathrm{g}}$ and their one-standard-deviation uncertainty bounds $\boldsymbol{\sigma}_{\delta}$. To calculate the vectors $\tilde{\tilde{\boldsymbol{\delta}}}_{\mathrm{g}}$, we used the element-wise product of the vectors $\tilde{\boldsymbol{\tau}}$ and $\tilde{\mathbf{c}}_{\mathrm{g}}$ (see Eq. (3.34) for the definition of $\tilde{\tilde{\boldsymbol{\delta}}}_{\mathrm{g}}$). In addition, Eq. (3.43) was used to calculate the vector $\boldsymbol{\sigma}_{\delta}$ because $\boldsymbol{\delta}_{\mathrm{g}}$ and $\boldsymbol{\delta}$ only differ in a constant (i.e. $\bar{\delta}_{\mathrm{g}}$ in Eq. (3.38)). Thus, their standard deviations are the same.

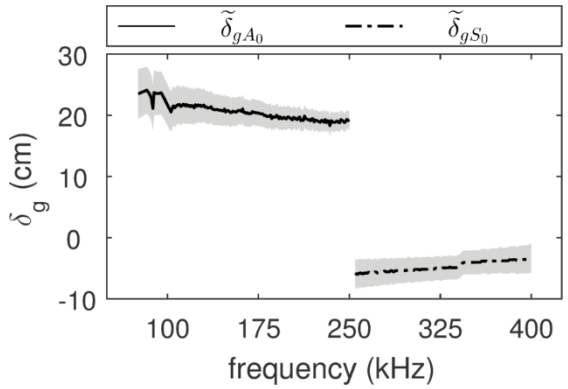

Figure 3.9: Product of group velocities and the time of first arrivals for the S_0 and A_0 modes; the shaded areas represent the standard deviations ($\boldsymbol{\sigma}_{\delta}$).

Figure 3.10 shows the values of the standard deviations $\boldsymbol{\sigma}_c$ and $\boldsymbol{\sigma}_\delta$ as well as the square root of the covariances, $\boldsymbol{\sigma}_{c\delta}$. In this figure, the vector $\boldsymbol{\sigma}_c$ is the uncertainty in the group velocities, which was calculated for Figure 3.8. To calculate the vectors $\boldsymbol{\sigma}_{c\delta}$ and $\boldsymbol{\sigma}_\delta$, Eq. (3.43) was used. In addition, the figure depicts the expected values of these vectors (i.e. the average values of them over their elements) (see Eq. (3.40)).

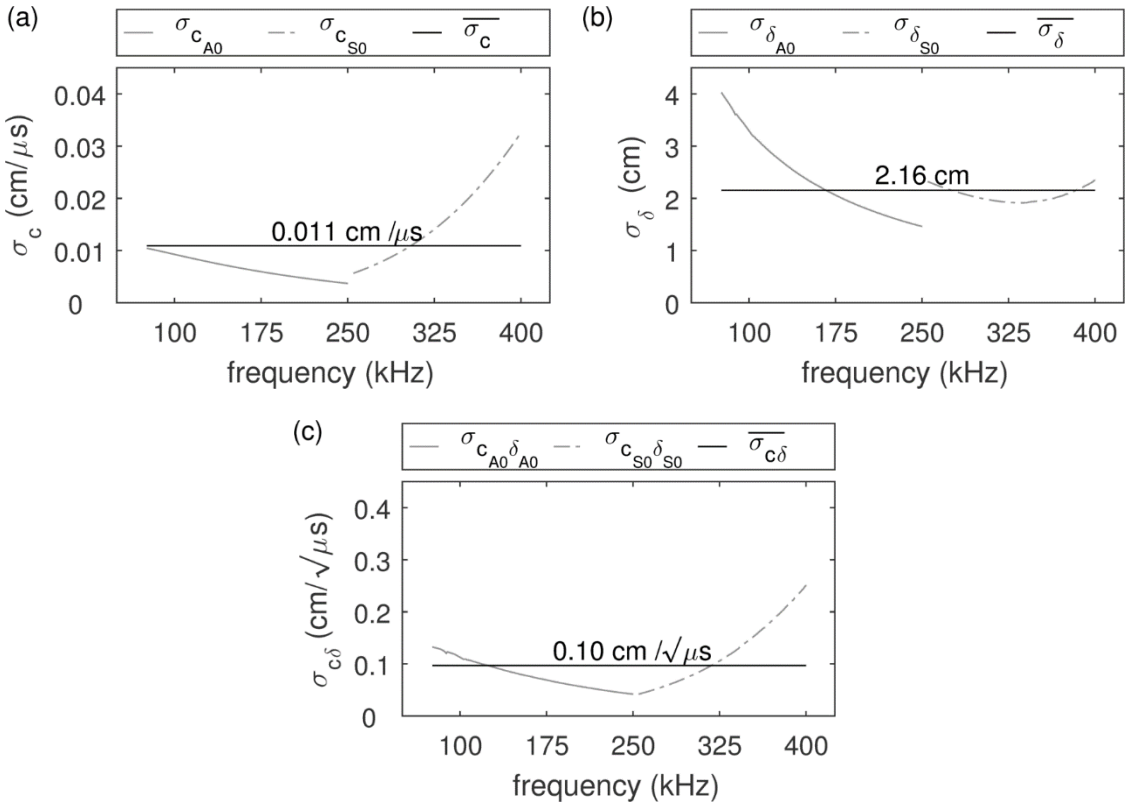

Figure 3.10: Averaging the TLS noise parameters over modes and frequencies: (a) the standard deviations of the group velocities, (b) the standard deviations of the parameter delta, (c) the square root of the covariance between the group velocities and the parameter delta.

Figure 3.11 compares the unbiased estimates $\hat{\mathbf{c}}$, $\hat{\boldsymbol{\delta}}$, and $\hat{\boldsymbol{\tau}}$ to their measured values $\tilde{\mathbf{c}}$, $\tilde{\boldsymbol{\delta}}$, and $\tilde{\boldsymbol{\tau}}$, respectively. At the lowest and highest frequencies, the differences between the measurements and estimates were the largest. To calculate the vectors $\hat{\mathbf{c}}$, $\hat{\boldsymbol{\delta}}$, Eq. (3.7)

was used after applying the TLS method. Then, we used Eq. (3.45) to calculate the unbiased estimate, $\hat{\tau}$.

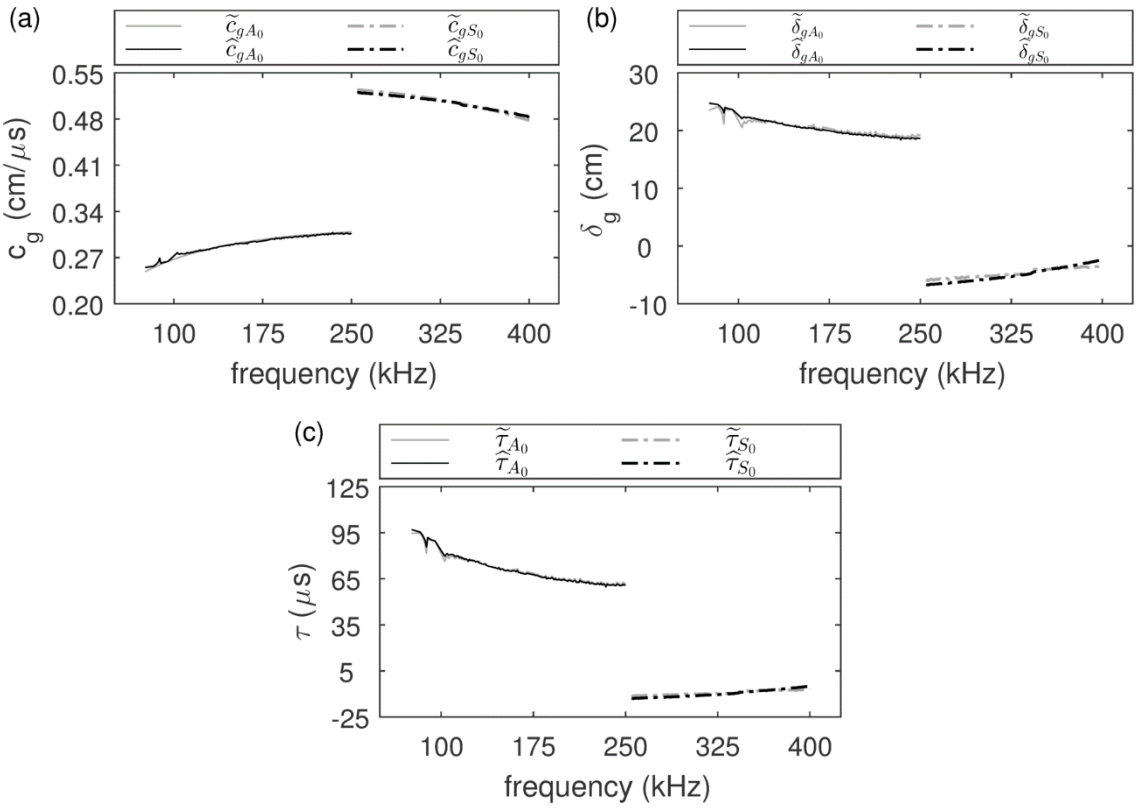

Figure 3.11: Comparison between the measured and estimated quantities; (a) groups velocity; (b) the product of group velocity and first arrival time; (c) first arrival time.

Figure 3.12 visualizes the output of the probabilistic source-to-sensor distance estimation (i.e. the first step of the framework) for the 96 PLB tests. It can be seen that the actual distances between the AE sensor and the location of the PLB tests (i.e. AE sources) are within the estimated uncertainty ranges. In these estimations, the minimum and maximum uncertainty ranges were 2.3 and 6.7 cm, respectively. In addition, it can be seen that the uncertainty ranges do not increase as the source-to-sensor distances increase. For example, the source with the maximum uncertainty was closer to the sensor than the source

with the minimum uncertainty. Therefore, the source-to-sensor distance estimation step of the framework can successfully estimate the distances and quantify their uncertainties.

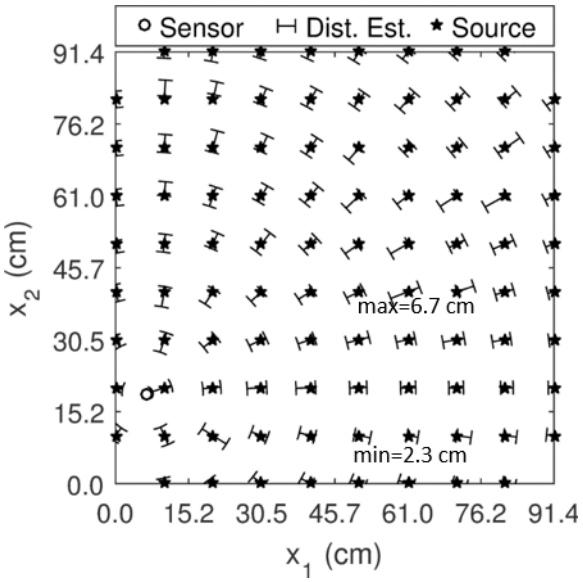

Figure 3.12: Direct distance estimations and their uncertainties.

3.6.2. Multipath (MP) model

Figure 3.13 visualizes the output of MP ray tracking algorithm for the AE source generated by a PLB test at $x_1 = 61.0$ cm and $x_2 = 30.5$ cm. The algorithm calculated 25 paths that less than or equal to three reflections occurs on them (i.e. $q = 25$ and $o_{\max} = 3$). For the sake of the figure's clarity, only some of these paths are shown here. The shortest distance (i.e. the direct source-to-sensor distance) was 55.8 cm, whereas the longest distance was 320.9 cm. On this longest path, three reflections occurred: from the top boundary, the bottom boundary, and again the top boundary.

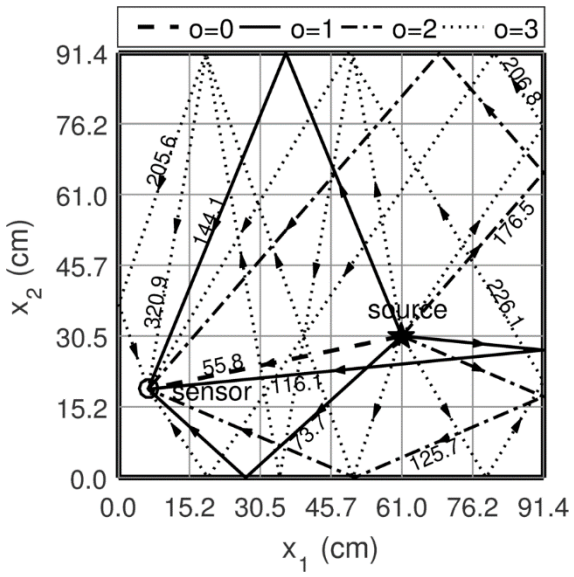

Figure 3.13: Output of MP ray tracking algorithm for up to three reflections; only some of the paths are shown; the numbers on the paths indicate their length in cm.

Figure 3.14 compares an analytically reconstructed AE envelope to its experimental envelope. These envelopes correspond to the PLB test performed at the coordinates $x_1 = 61.0$ cm and $x_2 = 30.5$ cm. To reconstruct the analytical envelope from the first arrival in the experimental signal, MP model was used. In this model, the γ coefficients used for S_0 and A_0 modes were $\gamma_{S_0} = 0.5$ and $\gamma_{A_0} = 0.8$, respectively. In this comparison, the experimental envelope was the modules of the CWT at the frequency of 250 kHz (i.e. $e(f,t) \mid f = 250$ kHz). As it can be seen in the figure, the two envelopes are more synchronous at the beginning. Although the correlation between the two envelopes was 90 percent in the time range [-200 μs, 600 μs], this correlation was 95 percent in the time range [-25 μs, 200 μs]. Such results indicate that MP model can reconstruct the patterns of the peaks and valleys of the filtered AE envelopes.

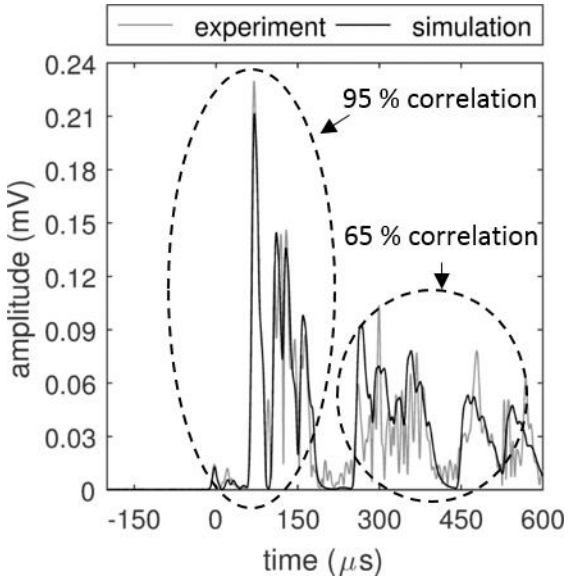

Figure 3.14: Envelopes of experimental and reconstructed AE signals for the PLB test at the coordinates of $x_1 = 61.0$ cm and $x_2 = 30.5$ cm.

3.6.3. Confidence contour estimation

Figure 3.15 visualizes the estimation of a confidence contour using 100 MC simulations. Specifically, these simulations were performed on the AE signal generated by a PLB test at the coordinates $x_1 = 61.0$ cm and $x_2 = 30.5$ cm. For each of these MC simulations, a correlation imaging with the resolution of 100 × 100 pixels was used (i.e. the grid spacing was $s_G = 0.914$ cm). Then, the pixel with the highest correlation was identified as the estimated source location in that MC simulation. As it could be seen in the figure, several MC points overlap at the same coordinates. Only 14 of these 100 MC points were unique, and the other 86 MC points overlapped with the unique points. In the figure, the size of a unique point indicates the number of overlapping MC points on it. According to the location of the MC points, the kernel density estimation (KDE) technique returns a two-dimensional probability density function (pdf). In this case, the calculated pdf had only one mode with a maximum of 0.27. To account for the diagonal distance between two

neighbor MC points, the bandwidth parameter of the KDE was selected to be $b_{KDE} = \sqrt{2}s_G$. Additionally, the figure shows the process of estimating the 95% confidence contour for this AE source. This contour is shown in white color. The height of this contour was selected such that the volume confined under the pdf and inside the contour to be 0.95. To calculate this volume, a two-dimensional trapezoidal integration rule was used. This numerical integration was performed on a grid with the spacing of $s_G / 3$.

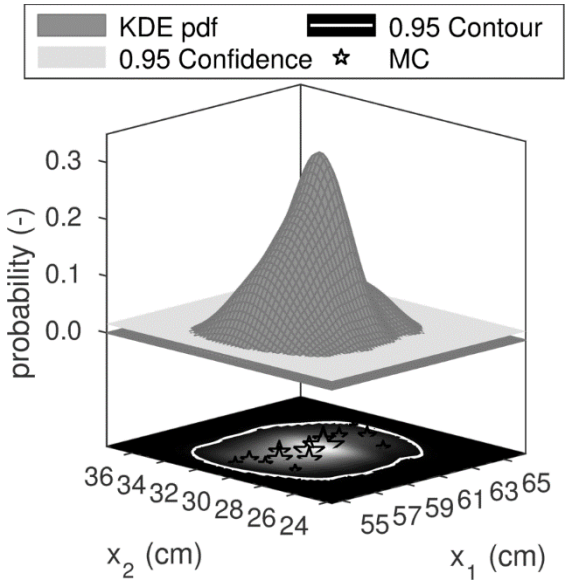

Figure 3.15: Confidence contour estimation using KDE on MC simulations.

Figure 3.16 shows the MC estimations of AE sources and their corresponding confidence contours. For the sake of clarity, only 24 out of the 96 PLB tests are shown. It can be seen in the figure that each confidence contour encloses its corresponding AE source. However, the confidence contours do not surround all MC points because the 95 percent confidence level requires an average of five MC points to be outside (these points may overlap). The farthest distance between a source and the centroid of its confidence contour was 5.3 cm. In addition, the shapes and sizes of the confidence contours vary with

the location of sources. While the shape of the largest confidence contour was similar to a 21.3-cm-long ellipse, a typical confidence contour was similar to a circle with the diameter of 11.7 cm. Moreover, the figure shows no blind zones in localizing the sources. Both the sources that were close to the edges of the plate and the sources that were close to the center of the plate were similarly localized. Therefore, the results prove that the developed framework can probabilistically estimate confidence contours for AE sources.

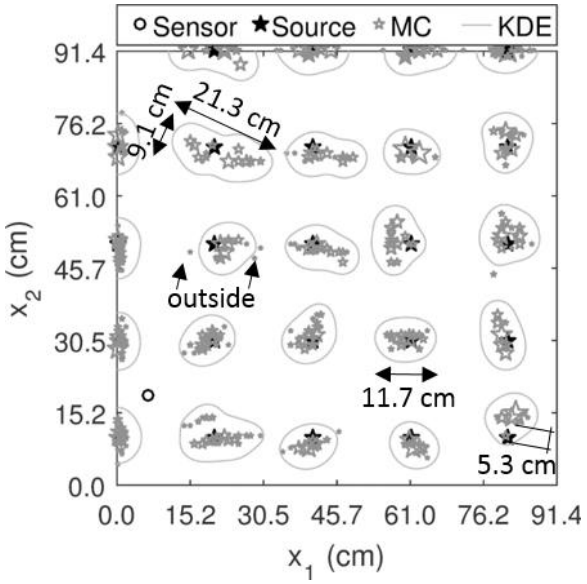

Figure 3.16: Confidence contours, MC points, and actual AE sources; for the sake of clarity, only 24 out of 96 PLB tests are shown.

3.6.4. Source localization in a noisy environment

Figure 3.17 visualizes localization confidence contours at six different environmental noise floors from 60 dB to 70 dB (at the steps of 2 dB). These confidence contours estimate the location of the PLB tests performed at the coordinates $x_1 = 61.0$ cm and $x_2 = 30.5$ cm. The figure shows that the size of the confidence contours monotonically grows as the noise floor increases. In these PLB tests, the average amplitude of the S_0 and A_0 mode were 60.5 dB and 80.1 dB, respectively. Such amplitudes correspond to the

signal-to-noise ratios (SNR) of 0.5, -1.5, …, -9.5 dB for the S_0 mode and 20.1, 18.1,…, 10.1 dB for the A_0 mode. It could also be observed from the figure that the confidence contours at the noise floor of 60 dB (SNR of 20.1 dB for the A_0 mode and 0.5 dB for the S_0 mode) converge to the noise-free confidence contour shown in Figure 3.16. In other words, below this noise floor, the systematic uncertainties become more dominant that the environmental noise. On the other end of the spectrum, the 70 dB noise floor was the maximum noise floor at which the proposed source localization framework was able to localize AE sources (SNR of 10.1 dB for the A_0 mode and -9.5 dB for the S_0 mode).

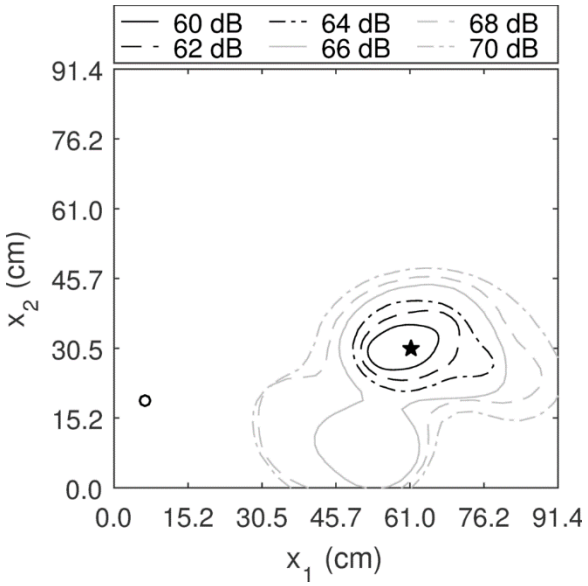

Figure 3.17: Confidence contours for the location of AE sources at different noise floors; the actual sources were simulated at $x_1 = 61.0$ cm, $x_2 = 30.5$ cm.

3.6.5. Overall performance and computation costs

Figure 3.18 summarizes the overall performance of the proposed framework over the 96 PLB tests performed in the noise-free condition. In particular, Figure 3.18(a) shows the histogram of the area confined within the confidence contours (see Figure 3.16). It can

be seen in the figure that 31 of the PLB were localized within confidence contours that their sizes were in the range of 65-95 cm^2 (in a bin with average value of 80 cm^2). The sizes of the smallest and largest of such uncertainty areas were 74.5 cm^2 and 204.9 cm^2, respectively. However, the dominant size among these uncertainty areas was in the range of 95-125 cm^2 (i.e. about one percent of the area of the plate). In addition, Figure 3.18(b) shows the histogram of the offset between the centroid of each uncertainty area and the corresponding actual source location. Although the maximum offset was 5.3 cm, the dominant value for this offset was 0.6 cm. Such results provide insight about the expected accuracy of single-sensor source localization.

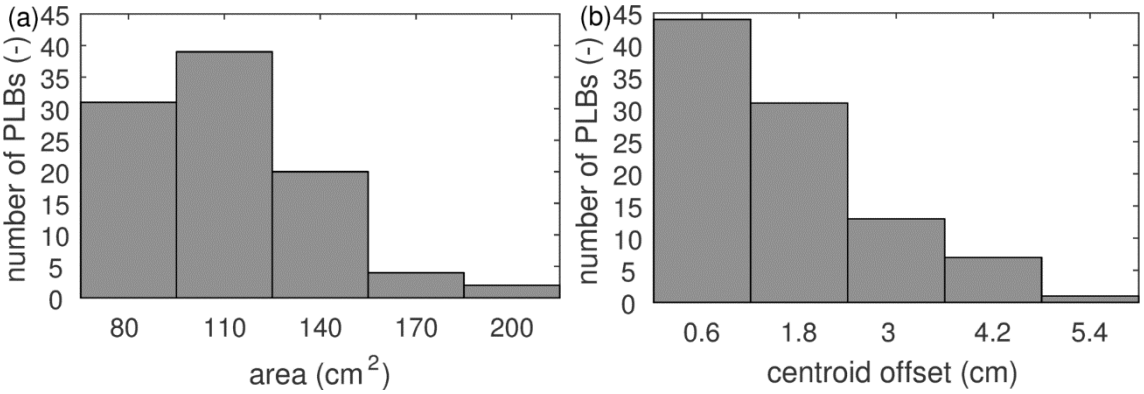

Figure 3.18: Histograms of 96 PLB localization results: (a) the size and (b) the offset of the uncertainty area.

The computation time of the proposed framework consists of the time spent on the probabilistic source-to-sensor distance estimation (step I) and MC simulations on the second and third steps of the framework. On average, a MATLAB implementation of the framework running on a core i5 (dual core) PC spends two seconds on the source-to-sensor distance estimation and 29 seconds on the Monte Carlo simulations. Specifically, steps II and III respectively take %90 and %10 of the time of each MC simulation. Since the MC simulations are suitable for parallel processing, the parallel implementation of the MC

simulation significantly decreases the computational costs. For example, on the same machine, the computation time decreased to 16 seconds. In addition, a further decrease is expected if one runs the algorithm on a graphical processing unit (GPU).

The abovementioned numbers are based on the 100 × 100-pixel resolution and 100 Monte Carlo simulations. It should be noted that the numbers do not include a one-time, three-minute calculation used in this implementation to populate a dictionary. The dictionary contains all propagation paths from each pixel to the sensor. This calculation time in not considered in the abovementioned numbers because once the dictionary is constructed, it is applicable to all future uses during the lifespan of the SHM system.

3.7. DISCUSSIONS

The proposed framework for probabilistic AE source localization has several advantages over the currently available algorithms. First, this framework uses only one omnidirectional sensor to localize AE sources and quantify the localization uncertainties. Any conventional AE sensor with broadband and omnidirectional characteristics could be used to work with the framework. In addition, this framework can monitor the entire area of a plate without any blind zones. As another advantage, this framework localizes AE sources in noisy environments. Based on such advantages, the proposed framework thus provides more comprehensive information regarding the location of sources.

3.8. CONCLUSIONS

The probabilistic localization of AE sources in metallic plates, as established with only one sensor in this study, is very important for the safety and performance of civil and aerospace structures. In real-world applications, the early localization of common defects, including fatigue cracks and corrosion dents, with the presented framework could prevent

catastrophic failures and reduce the life-cycle costs of the structures. In addition, the probabilistic approach of this paper enables decision makers to quantify localization uncertainties and repair damages at early stages.

This framework consists of three steps. In the first step, we used the total least squares (TLS) regression to estimate the source-to-sensor distances. In addition, we used the unscented transform (UT) and Cramer-Rao bound (CRB) to quantify lower bounds on the uncertainty of these estimations. In the second step, we used the estimated source-to-sensor distances in our approximate analytical model, called multipath (MP), to reconstruct edge-reflected late arrival envelopes based on their experimentally collected first arrivals. Finally, in the third step, we used correlation imaging to localize the sources. Additionally, Monte Carlo (MC) simulations were used to probabilistically propagate the quantified uncertainties from the first step to this last step. To estimate a confidence contour for these propagated uncertainties, we used the kernel density estimation (KDE) technique. Hsu-Nielsen pencil lead break (PLB) tests were performed on the surface and the edges of an aluminum plate to validate the proposed framework, and the results were compared with the actual source locations.

Despite the significant results presented in this study, a few limitations are identified. First, we conducted the experiments in a laboratory setting with controlled environmental conditions. Therefore, future research should account for uncertainties in temperature variations and conduct on-field experiments. In addition, this study does not consider uncertainty in material properties. Consequently, future studies should also investigate this source of uncertainty and quantify its contribution to the overall uncertainty.

The overarching goal in the future of this study is to monitor real structures that may not have free edges. For example, in airplane fuselages, stiffeners and stringers divide the skin plate into several bounded panels. Such panels also have the potential to be probabilistically monitored with only one AE sensor because their boundaries reflects the acoustic waves. However, reflection models need to be developed for stiffeners and stringers before one can explore this potential. As an option, pre-calculated finite element results could be augmented to the MP model. Therefore, future research should additionally focus on adopting this framework to more sophisticated structures and conducting experiments on real-world structural components. Besides, the proposed framework has the potential to be implemented in dual (active/passive) SHM systems(Nasrollahi et al. 2017), micro electromechanical systems (MEMS) (Kabir et al. 2015a; b), and wireless networks (Zahedi and Huang 2014). The future studies could also investigate this potential and use it to expand the applicability of this work.

3.9. ACKNOWLEDGMENTS

This work was supported by the Office of Naval Research # N00014-17-1-2367, Program Director Dr. Ignacio Perez, and National Science Foundation under the Grant # CMMI-1333506.

3.10. ADDITIONAL MATERIALS: FIRST ARRIVAL DETECTION AND ISOLATION

The Akaike information criterion (AIC) can be used to measure the first S_0 arrival time. However, this technique is not applicable to the first A_0 arrival time. To measure the first A_0 arrival time, a four-step, threshold-based procedure is proposed (see Figure 3.19). First, the signal is smoothed with a moving average filter. A post-processing threshold is then defined at a value proportional to the peak value of the signal's envelope. In the third

step, a half sign is fitted to the envelope in the range that starts at its first intersection with the post-processing threshold (i.e. point 8) and ends at the first immediate local maximum after the threshold crossing (i.e. point 9) . Finally, the half sign is extrapolated until it intersects the horizontal axis at point 7. We define the time of this zero crossing as the arrival time of the first A_0 packet.

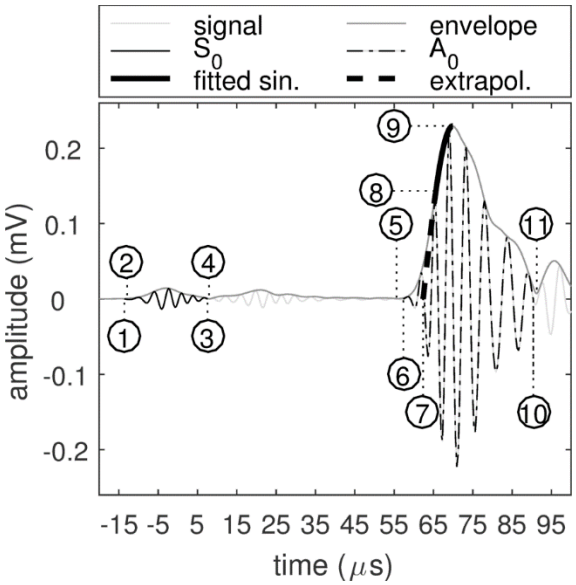

Figure 3.19: The process of measuring the first A_0 arrival time and isolating the first S_0 and A_0 packets.

The measured time of arrivals can be used to identify and isolate the first arrival packets. To isolate the first S_0 packet, we define a range starting from the minimum of the AIC (i.e. point 2) and ending at the first local minimum of the signal's envelope (i.e. point 4). Similarly, the first A_0 packet is defined in the range between the two local minima of the signal's envelope that are immediately before and after the measured time of arrival (i.e. points 5 and 11, respectively). The first and last zero crossings of the signal within these ranges define the first arrival packets. The only exception is the start point of S_0 packets, which is defined to be the last zero crossing of the signal before the time at which

the AIC is minimum (i.e. point 1). For the A_0 packet shown in the figure, these zero crossings are point 6 and 10, respectively.

Since the first arrival detection and isolation approach is applied on the real-part of the continuous wavelet transform (CWT) coefficients, it can handle the following conditions: a) noisy signals and b) signals with a higher amplitude for the S_0 mode than the A_0 mode (such as pencil lead break (PLB) tests performed on the edges of the plate in the middle of its thickness). In such cases, an attempt to measure the time of arrivals may fail at some frequencies but succeed at others. Therefore, the first arrival measurement approach only feeds the successful measurements to the probabilistic framework. For example, in the case of PLB tests on the edges of a plate, only the successful first arrival measurements of the A_0 mode along with the S_0 measurements are feed to the framework. This is because of the assumed higher amplitudes for the A_0 mode than the S_0 mode. Therefore, such A_0 measurements are only successful at low frequencies where this assumption is valid.

PAPER 4

Single-Sensor Acoustic Emission Source Localization in Plate-Like Structures Using Deep Learning

Arvin Ebrahimkhanlou and Salvatore Salamone[1]

4.1. SYNOPSIS

This paper introduces two deep learning approaches to localize acoustic emissions (AE) sources within metallic plates with geometric features, such as rivet-connected stiffeners. In particular, a stack of autoencoders and a convolutional neural network are used. The idea is to leverage the reflection and reverberation patterns of AE waveforms as well as their dispersive and multimodal characteristics to localize their sources with only one sensor. Specifically, this paper divides the structure into multiple zones and finds the zone in which each source occurs. To train, validate, and test the deep learning networks, fatigue cracks were experimentally simulated by Hsu–Nielsen pencil lead break tests. The pencil lead breaks were carried out on the surface and at the edges of the plate. The results show that both deep learning networks can learn to map AE signals to their sources. These results demonstrate that the reverberation patterns of AE sources contain pertinent information to the location of their sources.

---

[1]This chapter have been previously published as: Ebrahimkhanlou, Arvin, and Salamone, Salvatore (2018). "Single-Sensor Acoustic Emission Source Localization in Plate-Like Structures Using Deep Learning." Aerospace, 5(2), 50, DOI: https://doi.org/10.3390/aerospace5020050
The publisher is open access, and the dissertation advisor has allowed the direct reproduction of this paper. This is a non-typeset version of the accepted manuscript except for the numbering of sections, figures, and equations that have been adapted from the original to satisfy the formatting requirements for a dissertation. The author of this dissertation was the lead investigator conduction the research. The author was responsible of developing theories, designing and conducting experiments, analyzing the results, and writing the journal paper. The coauthors contributed by providing editorial comments.

Keywords: acoustic emission, guided ultrasonic waves, deep learning, autoencoders, convolutional neural networks, machine learning, edge reflection, reverberation patterns, structural health monitoring

4.2. INTRODUCTION

Metallic plate-like structures are ubiquitous in the aerospace industry. These structures are susceptible to different types of damage, including fatigue cracks and corrosion dents. Numerous studies exist in the literature concerned with the structural health monitoring (SHM) of metallic plates (Boller et al. 2009; Nicolas et al. 2016; Wilson and Goldfein 2017). Among these studies, SHM techniques based on acoustic emissions (AE) can detect and localize damage in metallic panels (Kundu 2014). For example, Kundu et al. (2007) developed an optimization-based approach for plates with known wave velocities and extended it later to localize sources in anisotropic plates with unknown properties (Kundu et al. 2015; Sen and Kundu 2018). The authors have also developed a dictionary-based source localization algorithm for simple isotropic plates (Dubuc et al. 2017). For more interested readers, Kundu (2014) has provided an in-depth review of AE source localization algorithms.

Despite the significant development of AE source localization algorithms, very few have been implemented in real structures. One reason for this lack of acceptance is the potential for these algorithms to emit false positives, which means either incorrectly identifying the location of defects or, even worse, localizing artificial defects that do not exist in the reality. One of the major sources of false positives in AE source localization is the large number of reflections and reverberations that appear in the tails (codas) of AE signals. This is because most localization algorithms rely only on the time difference of "first arrivals" in

AE waveforms and do not account for the reflections and reverberations generated by geometric features, such as boundaries, joints, stiffeners, and fasteners. In the literature, Hamstad et al. (2001, 2003) and Prosser et al. (1999) have extensively worked on numerical simulations of edge-reflected AE and have demonstrated how such reflections affect AE waveforms. In addition, Farhangdoust et al. (2017) worked on numerical simulations for stiffened rectangular plates. Alternatively, other researchers have taken an experimental approach (Bhuiyan et al. 2018; Carpenter and Gonnan 1995). In particular, Carpenter and Gonnan (1995) reported AE waveform in an aluminum (7075-T651) stiffened wing panel subject to cyclic fatigue tests and eventually yielding.

A common way to overcome the sophistications imposed by reflections and reverberations is to use many sensors and limit the localization to the area covered by the sensors. However, this approach can significantly increase the complexity and cost of the AE-based SHM system. Instead, some researchers have leveraged the additional information conveyed by the reflections and reverberations to improve the localization accuracy. For example, Achdjian et al. (2014) used a statistical approach to localize AE sources in a simple aluminum plate. In particular, they used the reverberation patterns of guided ultrasonic waves recorded by at least three sensors. Ernst et al. (2016) took a finite element approach to find the location of AE sources by propagating backward the waveforms recorded by a laser Doppler vibrometer. This approach required six hours of computation for each source localization.

Another major source of false positives in AE source localization is the multimodal and dispersive characteristics of AE waveforms. In thin plate-like structures, AE sources excite guided ultrasonic waves, specifically the Lamb waves. In fact, one can hardly find any AE application in plate-like structures that is not based on the Lamb waves. The problem is

that most AE source localization algorithms use either of the fastest propagating Lamb wave mode (first symmetric mode, S_0) or the higher amplitude mode (depending on the source type it could be either of the first symmetric mode, S_0, or the first anti-symmetric mode, A_0). As a result, they ignore the multimodal, and, in some cases, dispersive characteristics of AE waveforms. In contrast, some researchers have leveraged such characteristics to reduce the number of sensors required for localization (Holford and Carter 1999; Jiao et al. 2008; Toyama et al. 2001). For example, Holford and Carter (1999) used the far-field separation of Lamb wave modes in a 50-meter long I-beam to estimate the source-to-sensor distance with only one sensor.

To reduce the number of sensors and enhance the accuracy of source localization algorithms, the authors have leveraged both the reverberation patterns of AE waveforms as well as their dispersive and multimodal characteristics (Ebrahimkhanlou and Salamone 2017b; c; e). In particular, they developed an analytical model named "Multipath ray tracking" (Ebrahimkhanlou et al. 2016b) to simulate the reverberation patterns of AE waveforms. Their model reconstructs AE waveforms based on experimentally recorded first arrivals. The authors used this model to localize AE sources in an isotropic plate with only a single AE sensor (Ebrahimkhanlou and Salamone 2017c). They later quantified the uncertainty of this single sensor AE source localization (Ebrahimkhanlou and Salamone 2017e). However, their work was applied to a simple plate without any stiffener or fastener. To extend the previous work to plates with geometric features (e.g., stiffeners, rivets, etc.) and unknown material properties, this paper proposes a new data-driven approach based on deep learning. The main idea is to use deep learning to directly learn the reverberation patterns, multimodal characteristics, and dispersive properties of AE waveforms from a set of previously collected AE data. The goal here is to use only one sensor and localize AE

sources within plates that have stiffeners and rivet connections (Ebrahimkhanlou and Salamone 2017a). In particular, this paper focuses on AEs that are due to a sudden change in the strain field around the rivet connections of plate-like structures. Such sudden changes could be due to the progression of fatigue cracks that tend to grow from rivet connections in metallic plate-like structures. Furthermore, the paper considers AE sources on the surface and at the edges of plate-like structures. To experimentally simulate such AE sources, this paper uses Hsu–Nielsen sources (Hsu 1977). Then, it uses such sources to train two deep learning algorithms and map the resulted AE waveforms to the location of their sources. In this study, the localization is zonal, which means the structure is divided into multiple zones, and the zone in which an AE source occurs is detected. For example, to localize the fatigue cracks that tend to grow from the rivet connections, the area surrounding each rivet may be defined as a zone. Besides zonal source localization, what sets the current paper fundamentally apart from the previous work by the authors (Ebrahimkhanlou and Salamone 2017c; e) is the use of data-driven methods (i.e., deep learning) as opposed to analytical models (i.e., Multipath ray tracking).

Deep learning is a data-driven approach that eliminates the need for extracting manually designed, application-specific features from data. In the context of AE, one example feature is the time of arrival, which is traditionally used in time difference of arrival (TDOA) methods for source localization (Kundu 2014). In other words, the end-to-end architecture of deep learning allows it to be directly applied to data (i.e., signals, images, etc.) rather features extracted from the data. In fact, deep learning automatically learns and extracts representative features from data. In this way, deep learning also achieves a better performance than the traditional feature-based algorithms (Goodfellow et al. 2016; LeCun et al. 2015). In the literature, artificial neural networks, which are feature-based machine

learning algorithms, have been applied to both AE source localization and characterization (Al-Jumaili et al. 2016a; Sharif-Khodaei et al. 2012). However, little-to-no research has used deep learning to localize AE sources in plate-like structures. It worth mentioning that deep learning has been recently used for AE-based fault diagnosis in gearboxes and bearings (He and He 2017; Jia et al. 2016; Li et al. 2016). Nevertheless, to the best of authors' knowledge, none of such studies has used deep learning for AE source localization. To fill this gap, this paper uses two types of deep learning networks for AE source localization: (1) stacked autoencoders (Vincent et al. 2008), and (2) convolutional neural networks (Krizhevsky et al. 2012). In particular, the networks are used to identify (classify) the zone at which an AE source is generated. Since deep learning requires training data, this paper focuses on embedded and permanently attached monitoring systems that are trained once before deployment.

The organization of the subsequent sections of the paper is as follows. First, Section 4.3 reviews the theoretical aspects of the deep learning approaches used in this study. Then, Section 4.4 applies the deep learning networks to the problem of AE source localization. Sections 4.5 and 4.6 respectively include the experimental setup used to train, validate, and test the deep learning networks as well as the results obtained from them. Finally, concluding remarks are provided in Section 4.7.

4.3. DEEP LEARNING ARCHITECTURES

Deep learning uses neural networks that have multiple hidden layers (Goodfellow et al. 2016; LeCun et al. 2015). Similar to traditional neural networks, deep learning networks consists of a series of learnable neurons that nonlinearly map inputs to outputs. However, the input to deep learning networks are raw signals and images rather features extracted

from them. In other words, such networks automatically learn the most meaningful features directly from the signals and images.

Since the additional hidden layers of deep learning networks significantly increase the number of their tunable parameters, several deep learning architectures have been developed to keep the training process manageable. This paper, in particular, briefly reviews stacked autoencoders (Vincent et al. 2008) and convolutional neural networks (Krizhevsky et al. 2012).

4.3.1. Stacked Autoencoders

Stacked autoencoders are deep neural networks that consist of multiple pre-trained layers (Goodfellow et al. 2016; Vincent et al. 2008). Each layer of such networks is trained as a part of another neural network, which is named an autoencoder. The following subsections define the layers of a stacked network of autoencoders in details and explain their training procedure.

4.3.1.1. Autoencoders

Autoencoders are neural networks that reconstruct their input at their output (Goodfellow et al. 2016). A typical two-layer autoencoder consists of an encoder layer and a decoder layer (see Figure 4.1). The first layer of such networks maps (encodes) the input to a lower dimensional space, and the second layer maps (decodes) this compressed representation of the input data back to the original input space. In this way, an autoencoder automatically learns a compressed representation of its input. This compressed representation is called "features". Since the input and output of an autoencoder are the same, this learning process is unsupervised, which means it does not need any labeled training data. In the context of

AE source localization, labels could be the source coordinates/zone associated with each waveform.

Stacked autoencoders feed the encoded features of an autoencoder to another autoencoder for further compression. Let $\mathbf{x}^{(i-1)}$ be the input to the *i*-th layer of a stacked network of autoencoders. Then, the encoder of the *i*-th autoencoder, maps $\mathbf{x}^{(i-1)}$ to a lower dimensional space $\mathbf{x}^{(i)}$:

$$\mathbf{x}^{(i)} = f(\mathbf{W}^{(i)}\mathbf{x}^{(i-1)} + b^{(i)}) \tag{4.1}$$

In this equation, $\mathbf{W}$ and b are the weights and the bias of the encoder, respectively. The values of $\mathbf{W}$ and b are determined during the unsupervised training process of an autoencoder. In this notation, scalars, vectors, and matrices are indicated by a lower case italic font, a lower case bold roman font, and an uppercase roman font, respectively. In Eq. (4.1), f is the activation function of the encoder. A typical activation function for autoencoders is a sigmoid function, which is used in this study:

$$f(x) = \frac{1}{1+e^{-x}} \tag{4.2}$$

The decoder of the *i*-th autoencoder, maps $\mathbf{x}^{(i)}$ to the original feature space of $\mathbf{x}^{(i-1)}$:

$$\hat{\mathbf{x}}^{(i-1)} = f(\mathbf{W}_{\mathrm{d}}^{(i)}\mathbf{x}^{(i)} + b_{\mathrm{d}}^{(i)}) \tag{4.3}$$

where $\mathbf{W}_{\mathrm{d}}$ and b_{d} are the tunable weights and bias of the decoder. In this equation, $\hat{\mathbf{x}}^{(i-1)}$ is the reconstructed version of $\mathbf{x}^{(i-1)}$.

To learn the weights and the bias of an autoencoder, a loss function needs to be minimized. In this paper, a mean squared error is used as the loss function of autoencoders:

$$E = \frac{1}{N}\sum_{n=1}^{N} \| \mathbf{x}_n^{(i)} - \hat{\mathbf{x}}_n^{(i)} \|^2 \tag{4.4}$$

where N is the number of training samples, and the subscripts indicate the sample number. To minimize the loss function, a scaled conjugate gradient algorithm could be used (Møller 1993).

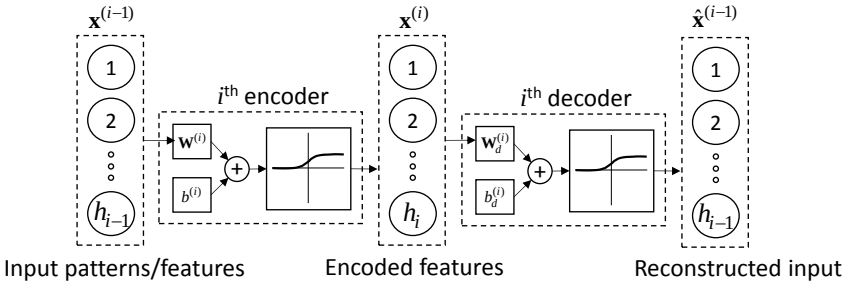

Figure 4.1: An autoencoder aims at reconstructing its input at its output.

4.3.1.1. Softmax Layer

A softmax layer is a single-layer neural network that classifies its inputs by mapping them into a finite number of output classes. This layer is typically the last layer of a stacked network of autoencoders that performs classification. For a network that consists of M autoencoders, the softmax layer can be represented mathematically as

$$\mathbf{y} = f(\mathbf{W}^{(M+1)}\mathbf{x}^{(M)} + b^{(M+1)}) \tag{4.5}$$

where $\mathbf{W}^{(M+1)}$ and $b^{(M+1)}$ are the weights and the bias of the softmax layer. The activation function of a softmax layer is named a "softmax function". This differentiable function is defined as

$$f(\mathbf{x}) = \frac{e^{\mathbf{x}}}{\sum e^{\mathbf{x}}} \tag{4.6}$$

in which the summation is over the elements of $\mathbf{x}$.

Unlike autoencoders, the training process of a softmax layer is supervised. In other words, this process requires a set of training samples, and for each of them (i.e., for each $\mathbf{x}_n^{(M)}$), the corresponding class $\mathbf{t}_n$ needs to be known. In this notation, all elements of the vector

$\mathbf{t}^{(n)}$ are equal to zero except for one of them that indicates the class membership. For example, if $\mathbf{x}_n^{(M)}$ belongs to the *j*-th class, only the *j*-th element of $\mathbf{t}^{(n)}$ is one. To train a soft max layer, a cross-entropy loss function is usually minimized (Bishop 2006):

$$E = -\sum_{n=1}^{N} (\mathbf{t}_n \cdot \ln \mathbf{y}(\mathbf{x}_n^{(M)})) \tag{4.7}$$

4.3.1.2. Fine-Tuning

Fine-tuning is the process of updating the weights and biases of an entire deep learning network. Fine-tuning is usually performed after all layers of the network are individually trained. In this process, the pre-trained values for the weights and biases are used as initialization for minimizing the cross-entropy loss function of the entire network:

$$E = -\sum_{n=1}^{N} (\mathbf{t}_n \cdot \ln \mathbf{y}(\mathbf{x}_n^{(1)})) \tag{4.8}$$

4.3.2. Convolutional Neural Networks

Convolutional neural networks are deep learning networks that take images as input. Unlike traditional neural networks that their layers are one dimensional, each layer of a convolutional neural network has three dimensions: width, height, depth (see Figure 4.2(a)). For example, the input layer of a network that takes color images has two dimensions for the width and height of the input images and the third dimension for its color channels. In the consecutive layers of a convolutional neural network, from the input layer to the output layer, the width and height of the layers gradually decrease, but their depth increases. This decrease continues in such a way that the width and height of the output layer are equal to one.

In convolutional neural networks, unlike traditional neural networks that each neuron is fully connected to all neurons in the previous layer, the neurons are only connected to a

small region in their previous layer. This region is named the "receptive field" of the neuron (see Figure 4.2). This architecture allows the number of tunable parameters (i.e., weights and biases) remain manageable even for large input images. This particular feature makes the training time of the convolutional neural networks less sensitive to the size of the input data than the staked autoencoders.

Figure 4.2(c) visualizes two parameters that control receptive fields: stride and zero-padding. In particular, the stride parameter defines the distance between the receptive fields of two neighbor neurons. Zero-padding is another parameter that controls the interactions of the receptive fields with the edges of the previous layer. Specifically, this parameter defines the number of added zeros to the edges of the previous layer. Both stride and zero-padding are non-tunable parameters that remain constant during the learning process.

Each layer of a convolution neural network consists of multiple channels (see Figure 4.2). The numbers of channels define the depth of a layer. For each channel, there is a dedicated image processing filter that its output defines the neuron values. Depending on the type of the layer, the filter performs different tasks. The following describes the most common types of layers uses in a convolution neural network.

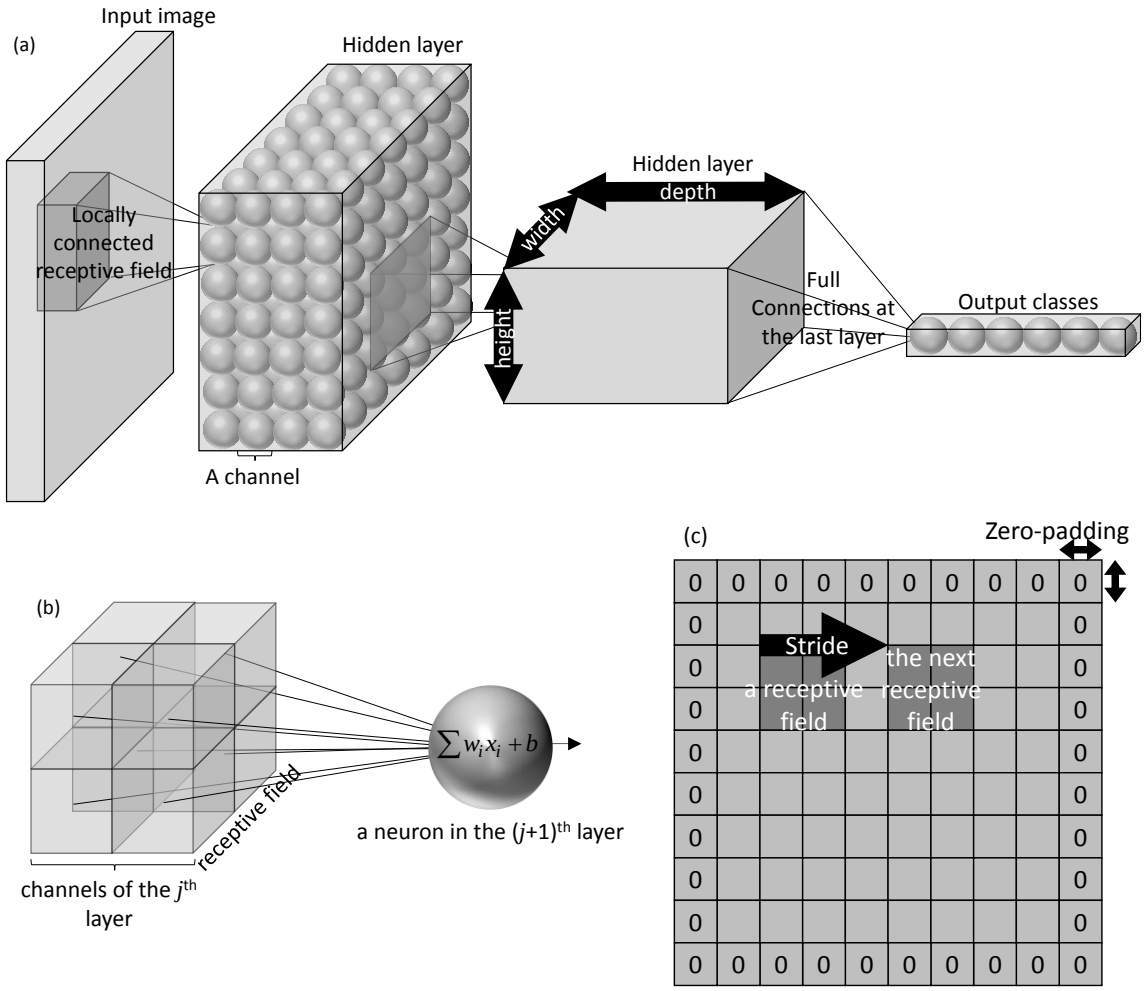

Figure 4.2: Convolutional neural networks: (a) conceptual architecture, (b) a neuron with a 2 × 2 receptive field on a two-channel layer, and (c) zero-padding and stride on a channel.

4.3.2.1. Convolutional Layer

Each channel of a convolutional layer applies the following filter to the receptive fields of its neurons:

$$\mathbf{x}^{(i)} = f(\mathbf{W}^{(i)}\mathbf{x}_r^{(i-1)} + b^{(i)}) \tag{4.9}$$

In this equation, $\mathbf{x}$, $\mathbf{W}$, b, and f are, respectively, the value of neurons as well as their weights, bias, and activation function. It is important to note that $\mathbf{x}$ here is a three-

dimensional data structure, in which the first two dimensions are the height and width of the image, and the third dimension is for the number of channels. In this notation, $\mathbf{x}_r^{(i-1)}$ is the receptive field of the *i*-th layer (see Figure 4.2(a)). A receptive field $\mathbf{x}_r$ is a subset of $\mathbf{x}$ in the first two dimensions, but as Figure 4.2(b) shows, it includes all the channels in the third (depth) dimension. The number of channels in a convolutional layer is another non-tunable parameter (other than the stride and zero-padding) that needs to be defined at the beginning and does not change during the learning process.

A commonly used activation function in most convolutional neural networks is a "rectified linear unit (ReLU)" (Glorot et al. 2011). A ReLU function nonlinearly maps each negative element x in the input vector $\mathbf{x}$ to zero:

$$f(x) = \begin{cases} x & x > 0 \\ 0 & otherwise \end{cases} \tag{4.10}$$

4.3.2.2. Max-Pooling Layer

Max-pooling layers have the same number of channels as their input layer. In this type of layer, the receptive fields are two dimensional and only apply to the height and width (not depth) of the corresponding channel in the previous layer. The neurons of a max-pooling layer calculate the maximum value in their receptive field:

$$\mathbf{x}^{(i)} = \max(\mathbf{x}_r^{(i-1)}) \tag{4.11}$$

Max-pooling layers usually use a stride value equal to two to reduce the height and width of their inputs. In this way, max-pooling layers down-sample their inputs.

4.3.2.3. Fully Connected Layer

A fully connected layer is one of the last layers in a convolutional neural network. This layer reduces the height and width of its inputs to one. The construction of a fully connected

layer is similar to a convolutional layer, but it has two main differences: (1) the receptive field of a fully connected layer has the same height and width as its input layer. (2) The neurons of a fully connected layer do not apply a ReLU function. For example, in a classification problem, which is the case in this paper, such neurons apply a "softmax" function instead (see Eq. (4.6)).

4.3.2.4. Visualizing the Inception of a Convolutional Neural Network

Images that strongly activate a specific channel in a layer of a convolutional neural network represent the inception of that channel. To generate such images, which are generally named "deep dream" images, one may use stochastic optimization and find an image that maximizes the activation (Mordvintsev, Alexander Olah and Tyka 2015). In particular, the optimization starts with a random noisy image and iteratively changes the image to increase the activation. In this process, a priori statistics constrain the optimization to produce images with similar statistics to natural images. For the final layer of convolutional neural networks, since each neuron corresponds to an output class, the deep dream images visualize the way that the entire network perceives that class.

Occluding a part of the input image is another way to visualize the inception of a convolutional neural network (see Figure 4.3) (Zeiler and Fergus 2014). This technique tests the performance of a network on images that are partially occluded. Then, the test is repeated after slightly moving the occlusion. In this way, this technique produces a map of areas in the image that are the most sensitive to the occlusion. This map visualizes specific areas within images that the convolution neural network relies upon to classify an image.

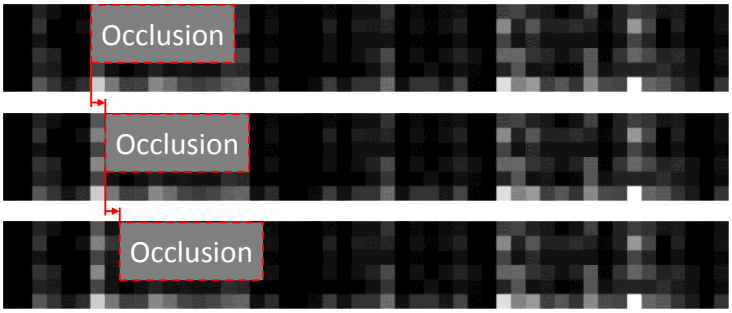

Figure 4.3: A moving occlusion window partially covers test images while the accuracy of the convolutional neural network is being evaluated.

4.3.3. Overfitting Mitigation in Training Deep Networks

Almost all machine learning algorithms, including neural networks, are susceptible to overfitting. Overfitting means that the network is so specialized to the training examples that it cannot generalize its input-output map to an unseen dataset. In this case, the network achieves a minimum loss function only for the training dataset rather any unseen data.

Several approaches exists to avoid overfitting: regularization, cross-validation, and dropout (Agarwal et al. 2011; Bishop 2006; Goodfellow et al. 2016). In particular, regularization ensures generalization by penalizing large weights and biases. Since such large values specialize the deep learning network to specific patterns, enforcing smaller weights and biases prevents overfitting to such patterns (Bishop 2006).

Cross-validation is another approach to avoid overfitting (Bishop 2006). This approach withholds a subset of training dataset, which is named the *"*validation set*"*, from the gradient descent algorithm. Cross-validation uses the value of the loss function on the validation set as the stopping criteria for the training procedure. Therefore, the gradient descent algorithm uses the training set to update the weights and biases of the network, but it stops when the loss function is minimized on the validation set.

Dropout is another regularization technique mostly used in convolutional neural networks. This technique adds an additional layer to the network. The added layer randomly ignores some of its input neurons during the training process (Agarwal et al. 2011). In other words, this technique forces other neurons to step in and make predictions instead of the missing neurons. In this way, the network becomes less sensitive to any specific neuron, and the dropout layer makes it less likely for the network to overfit to the training data. To be more effective, the dropout layer is usually applied before the fully connected layer of a convolution neural network.

4.4. ACOUSTIC EMISSION SOURCE LOCALIZATION WITH DEEP LEARNING

This study uses deep learning to identify the zone in which AE occurs. To achieve this goal, deep learning uses a set of AE waveforms and their corresponding source locations as training examples and constructs a nonlinear map between the waveforms and the source locations. In this process, deep learning leverages the reverberation patterns as well as the multimodal and dispersive characteristics of AE waveforms to determine their source location.

Deep learning has the advantage of leaning directly from signals and images, rather features extracted from them. In the context of AE source localization, this eliminates the need to extract features, such as time of arrival. To leverage multimodal and dispersive characteristics of AE waveforms, this study applies deep learning directly to a time-frequency transform of AE waveforms. In particular, a continuous wavelet transform is used.

The wavelet transform is widely used in various structural health monitoring applications (Ebrahimkhanlou et al. 2016a; Mohammadi-Ghazi et al. 2018; Mostavi et al. 2017; Sarrafi

et al. 2017, 2018b; a; Sarrafi and Mao 2018). Let $r(t)$ be an input signal. The wavelet coefficients are defined as:

$$C(f,\tau) = \frac{1}{\sqrt{s(f)}} \int_{-\infty}^{+\infty} r(t)\Psi^*(\frac{t-\tau}{s(f)})\mathrm{d}\tau \tag{4.12}$$

In this equation, τ is the translation parameter, s is the non-dimensional scale parameter defined as $s(f) = f_c \cdot f_s / f$, and Ψ^* is the complex conjugate of the mother wavelet $\Psi(t)$. This study uses a complex Morlet mother wavelet:

$$\Psi(t) = \frac{1}{\sqrt{\pi f_\mathrm{b}}} \exp(2\pi f_\mathrm{c} jt - \frac{t^2}{f_\mathrm{b}}) \tag{4.13}$$

The following subsections explain how stacked autoencoders and convolutional neural networks may use the continuous wavelet coefficients of AE waveforms to localize AE sources.

4.4.1. Stacked Autoencoders

The input to stacked autoencoders is a one-dimensional signal. To construct a multi-frequency representation of AE waveforms, this paper converts the modulus of the wavelet coefficients to a one-dimensional pattern (see Figure 4.4). In particular, a few frequencies are selected from the most dispersive and high-amplitude frequency range of AE waveforms. In this study, this range approximately starts from 25 kHz and ends at 500 kHz. While selecting more frequencies will feed more information to the stacked autoencoders, it will also increase the size of the input pattern and hence the computation time. To balance this trade-off, this study uses three frequencies. Specifically, 75 kHz, 200 kHz, and 325 kHz are selected to respectively represent the low-, mid-, and high-frequency contents in this range. Then, a fixed 500 μs-long window (starting from −10 μs to 490 μs) is used to resample the moduli of the wavelet coefficients. The size of the temporal window is

selected in such a way that includes multiple reflections from the geometric features of the structure. While a longer window will include more reflections, it will increase the size of the input pattern to deep learning networks and hence the computation time. In other words, here there is another tradeoff between the accuracy and computation time, which, in this case, is balanced with a 500 μs-long window. The resampling is performed at frequencies that are as twice as the three wavelet frequencies (i.e., 37.5 kHz, 100 kHz, and 162.5 kHz, respectively). Finally, the concatenation of the resampled data points constructs the input to the deep learning network. In this study, this multi-frequency representation of AE waveforms consists of 149 data points that are normalized to have the maximum value of one (see Figure 4.4(b)).

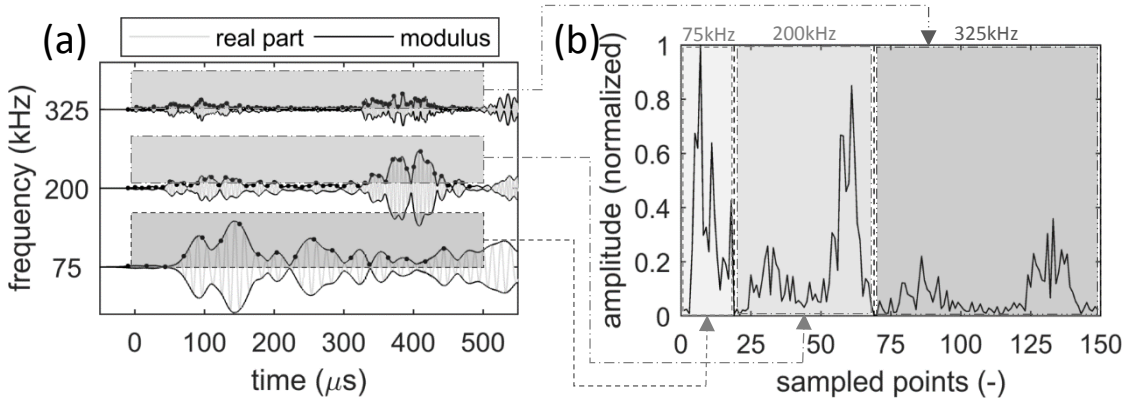

Figure 4.4: Across-frequency resampling of the modulus of the wavelet coefficients at 75 kHz, 200 kHz, and 325 kHz: (a) the real part and modulus of the wavelet coefficients for an AE waveform (resampling is indicated with dots); (b) the multi-frequency representation of the AE waveform.

The stacked autoencoders used in this study consists of two autoencoders and a softmax layer (see Figure 4.5). In this network, the autoencoders compress the input patterns first into 40 encoded features and then into 15, further compressed features. Since the goal is to identify the zone in which simulated fatigue cracks generate AE, a softmax layer is used to

localize the AE source. The input to the softmax layer is the encoded features by the second autoencoder, and the output is the zone number. In other words, this layer classifies AE waveforms into a finite number of classes that each correspond to a zone of the structure. Specifically, the output is a vector that the values of its elements are negligible except for one of them, which indicates the zone number. As Figure 4.5 shows, the number of zones is indicated by the parameter Z.

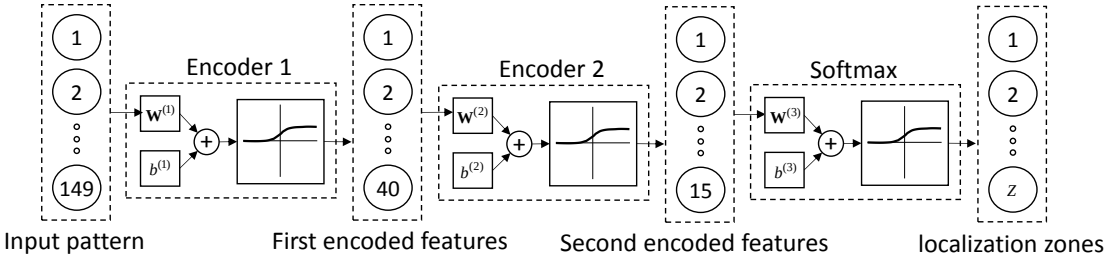

Figure 4.5: A stack of two autoencoders and a softmax layer. This architecture is used to find the zone in which an AE source occurs.

4.4.2. Convolutional Neural Networks

The input to convolutional neural networks is a multi-dimensional signal, which is usually a two-dimensional image. This allows directly using the modules of the wavelet coefficients as input to this deep learning network. As Figure 4.6 shows, this study normalizes the moduli of wavelet coefficients and converts it to an input image. In this process, wavelet coefficients are calculated at multiple frequencies, which are selected from the most dispersive and high-amplitude frequency range of AE waveforms (in this study, 25 kHz to 500 kHz). Since convolutional neural networks are less sensitive to large inputs than stacked autoencoders, this study uses six frequencies. As discussed in Section 4.3.2, this flexibility is due to the deployment of "receptive fields" in convolutional neural networks. In particular, the six frequencies start with 75 kHz and with the steps of

75 kHz end at 450 kHz. These six frequencies constitute the vertical axis of the input image. In the time domain, the horizontal axis of the input image corresponds to the same time window used for the stacked autoencoders (−10 μs to 490 μs). However, for this axis, the modules of the wavelet coefficients are calculated every 10 μs, which makes the size of the input image 6 × 50.

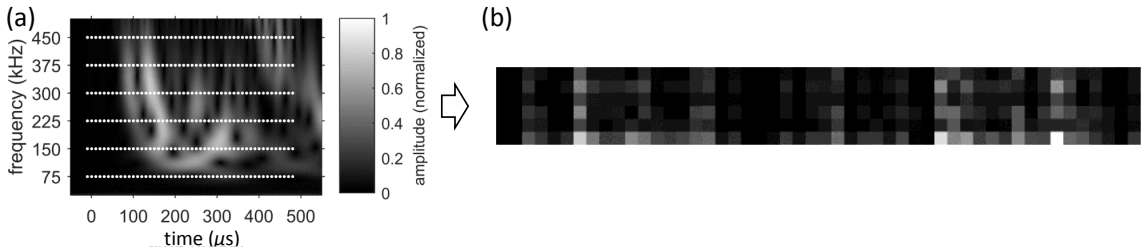

Figure 4.6: The process of producing the input image to the convolutional neural network: (a) a normalized continues wavelet transform, sampled pixels are overlaid, (b) the constructed image.

The convolutional neural network of this study has three convolutional layers and two max pooling layers (see Figure 4.7). The two types of layers are alternatively arranged to gradually reduce the height and width of the images while increasing their depth (i.e. the number of channels). The last convolutional layer is followed by a dropout layer followed by a fully connected layer and a softmax layer. The purpose of these layers is to prevent overfitting, dimensionality reduction, and classification, respectively.

All convolutional layers pad a zero pixel in all four directions (i.e., left, right, top, and bottom). Since the height and width of all convolutional filters is 3 × 3, this zero-padding ensures that the height and width of images remain the same before and after the convolutional layers. However, the number of the filters used in the convolutional layers gradually increase the depth of images (i.e., their number of channels). In particular, the convolutional layers increase the depth from 1 to 16, and then to 32, and eventually to 64.

Max pooling layers down-sample the images. In particular, a two-pixel stride reduces the height and width of the images by half. However, the depth of the images remains the same before and after the max pooling layers. In this study, no zero-padding is used for the max pooling layers.

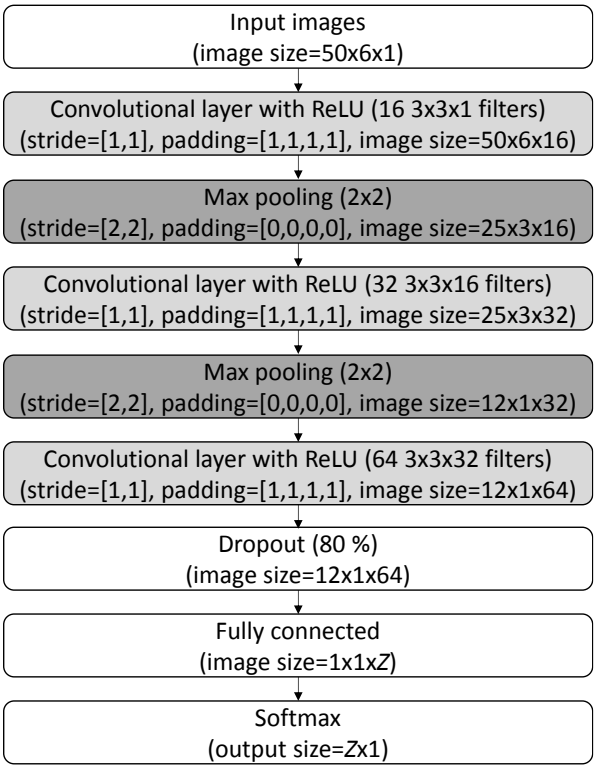

Figure 4.7: The architecture of the convolutional neural network used to discriminate AE sources. The parameter here indicates the number of zones.

Unlike convolutional and max pooling layers, as Figure 4.7 shows, the dropout layer does not change the size of the images. However, the fully connected layer reduces their width and height to one while matching the depth with the number of output classes. Since this study performs zonal localization, the number of output classes are the same as the number of considered zones. To find the neuron with the highest activation in the fully connected

layer, the network uses a softmax layer as the last layer. This layer classified the AE waveforms into multiple classes the each correspond to a zone of the structure.

Convolutional neural networks can nonlinearly map their inputs to their outputs. The sources of nonlinearity in the network used in this study can be classified into three groups: (1) the rectified linear units (ReLU) that are used as the activation function of the convolutional layers, (2) the down-sampling performed in the max pooling layers, and (3) the final softmax layer.

4.5. EXPERIMENTS

Figure 4.8 shows the experimental setup used to evaluate the effectiveness of the proposed deep learning approaches. In particular, the specimen was a 6061-T6 aluminum plate (914.4 mm × 914.4 mm × 3.2 mm). To simulate a realistic plate-like structure with a stiffener, a one-inch-wide aluminum strip (the same material and thickness) was fastened to the back of the plate with five ¼" (6.35 mm) rivets (see Figure 4.8(b)). The rivets are numbered in Figure 4.8(a). To collect AE waveforms, the plate was instrumented with only one AE sensor (PICO, Physical Acoustics Corporation) (see Figure 4.8(a)). The main reason to select the PICO sensor was its broad-band frequency response. Based on the recommendation of previous studies, the location of the sensor was selected away from the symmetry lines of the plate (Ebrahimkhanlou and Salamone 2017c; e). Specifically, it was attached at the coordinates (63.5 mm, 190.5 mm) relative to the lower left corner of the plate. It is important to mention that the sensor could have been affixed to any other location on the plate if it had been offset against symmetries. However, symmetric locations, such as the center of the plate, should be avoided. To fix the sensor in its place, hot glue was used.

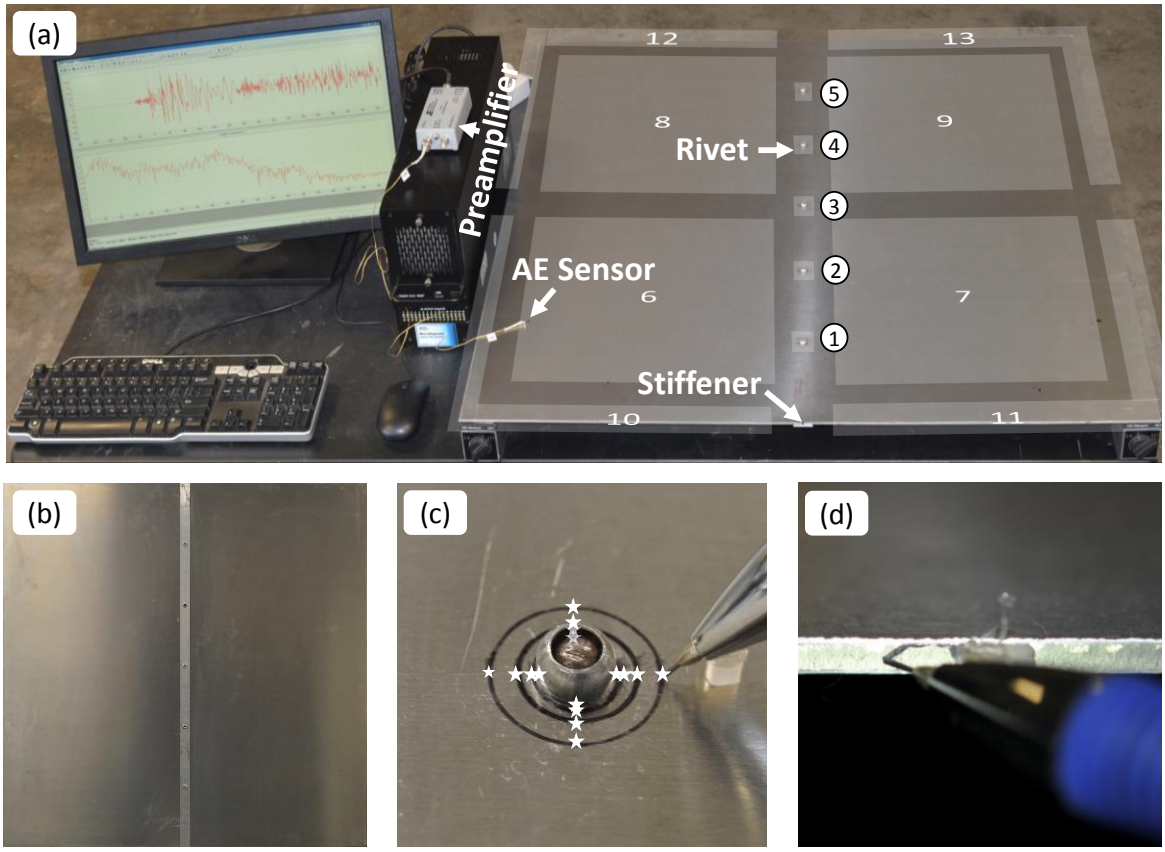

Figure 4.8: (a) Experimental setup, thirteen zones are labeled in the image; (b) stiffener on the back of plate; (c) a pencil lead break test next to a rivet (zones 1–5); (d) a pencil lead break test at the edge of the plate (zones 10–13).

In order to simulate fatigue cracks that usually initiate from rivets and fastener holes, 416 Hsu-Nielsen pencil lead break tests were performed (Hsu 1977). In particular, thirteen zones were considered on the plate (see Figure 4.8(a)), and 32 Hsu–Nielsen sources were simulated in each. In the first five zones, as Figure 4.8(c) shows, 32 AE sources were simulated next to each rivet connection. Specifically, the tests included the four sides of each rivet (right, left, top, and bottom) and four distances from the edge of each rivet: 0.8 mm, 1.6 mm, 3.2 mm, and 6.4 mm. For each of these 16 combinations, the Hsu–Nielsen tests were repeated twice. Zones six to nine were square areas that had a two-inch clearance from the edges and the center lines of the plate. In each of these four zones, 32 Hsu–Nielsen

AE sources were simulated. In each zone, simulations were performed at a 4 × 4 grid and at each grid intersection, the tests were repeated twice. In zones ten to thirteen, Hsu-Nielson sources were simulated at the edge of the plate. In each zone, 32 pencil lead break tests were performed at sixteen locations. Specifically, the tests were spaced by a two-inch distance, and each test was repeated twice. In terms of data acquisition, the AE signals were first amplified by 40 dB and then filtered by a 5 kHz–1 MHz band-pass analog filter before being digitized at the sampling frequency of 5 MHz. In addition, during the post-processing in MATLAB, a digital band-pass filter (Butterworth) was used to limit the frequencies to 25 kHz–500 kHz. Finally, the AE waveforms and their corresponding zone number were randomly divided into training, validating, and testing sets. In particular, 80%, 10%, and 10% of the data was used for training, validation, and testing, respectively. In this study, two scenarios were considered: (1) AE sources only in the first five zones (at the vicinity of rivet connections) and (2) and AE sources in any of the thirteen zones. In the first experimental scenario, the data set includes 160 simulated sources: 120 for training, 16 for validation, and 16 for testing. In the second the data set includes all 416 Hsu-Nielsen sources: 332 for training, 42 for validation, and 42 for testing. To allow comparison between the stacked autoencoders and the convolutional neural network, in each experimental scenario, a similar randomization was used to divide the data into training, validation, and testing sets. However, the randomization used for the first and second scenarios are different because the size of the two databases is different.

4.6. RESULTS

This section presents the results obtained from the stacked autoencoders and the convolutional neural network. For the first experimental scenario, the results obtained

during the training phase of both deep learning networks as well as final zonal localization results are presented. In this experimental scenario, the focus is only on the first five zones. These zones correspond to the AE source simulated near the five rivet connections. Figure 4.9 shows samples of AE waveforms used to train, validate, and test the two deep learning approaches. Deep learning leverages the difference between AE waveforms to define a map between them and their corresponding source location. For the second experimental scenario, only the final localization results and the required computational time are discussed.

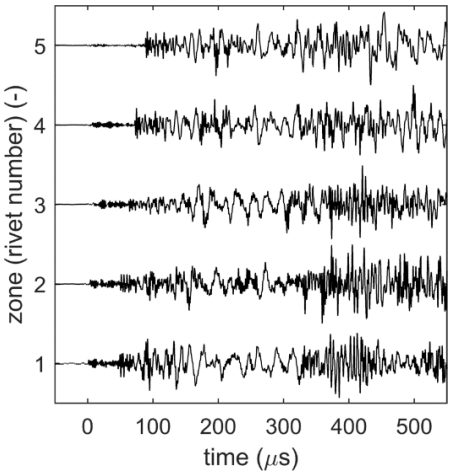

Figure 4.9: Samples of AE waveforms simulated at the rivets (the first five zones). To allow comparison, the waveforms were normalized.

4.6.1. Stacked Autoencoders

Figure 4.10 visualizes the learning curves of the stacked autoencoders used in the first experimental scenario of this study. In particular, Figure 4.10(a-b) correspond to the unsupervised training of the first and second autoencoders, Figure 4.10(c) belongs to the supervised training of the softmax layer, and Figure 4.10(d) corresponds to the supervised fine-tuning of the entire deep learning network. All four cases use a scaled conjugate

gradient algorithm (Møller 1993) for training. While the autoencoders use a mean square error as their loss function (see Eq. (4.4)), a cross-entropy loss function was minimized for the softmax layer and the entire network (see Eq. (4.7) and Eq. (4.8)). To avoid overfitting, a weight decay regularization term was added to the loss functions. In addition, in all four cases, the optimization was stopped when the global minimum was achieved on the validation data. In the graphs, the global minimum is indicated as the "best" results. For example, the global minimum for the first autoencoder was reached after 448 steps of training.

Figure 4.11 shows the input and output of the autoencoders for a randomly selected pattern from the testing dataset of the first experimental scenario. Since this specific pattern was never used in the training and validation processes, it is considered new data. Figure 4.11(a-b) respectively correspond to the first and second autoencoders. While the first autoencoder reconstructs the 149 input patterns, the second autoencoder reconstructs the 40 encoded features by the first autoencoder. Both figures demonstrate that the two autoencoders can successfully reconstruct their input. In addition, Figure 4.11(c) compares the input patterns with their reconstructed version by the combination of the two autoencoders. This figure, in particular, also demonstrates negligible information loss after two layers of encoding and decoding. To produce the reconstruction plot, the output of the second autoencoder was decoded by the first autoencoder.

Figure 4.12 contains localization results obtained from the stacked autoencoders used in the first experimental scenario. In particular, the deep learning network was tested on 16 randomly selected Hsu–Nielsen pencil lead break tests. These 16 waveforms consist of respectively five, one, three, three, and four AE sources at the first to fifth rivet connections. The confusion matrix shows that the network successfully identified the corresponding

rivets to all 16 tests. For example, the first entry of the matrix reads as the testing subset included five randomly selected AE sources that were simulated at the first rivet and all five of them were correctly localized.

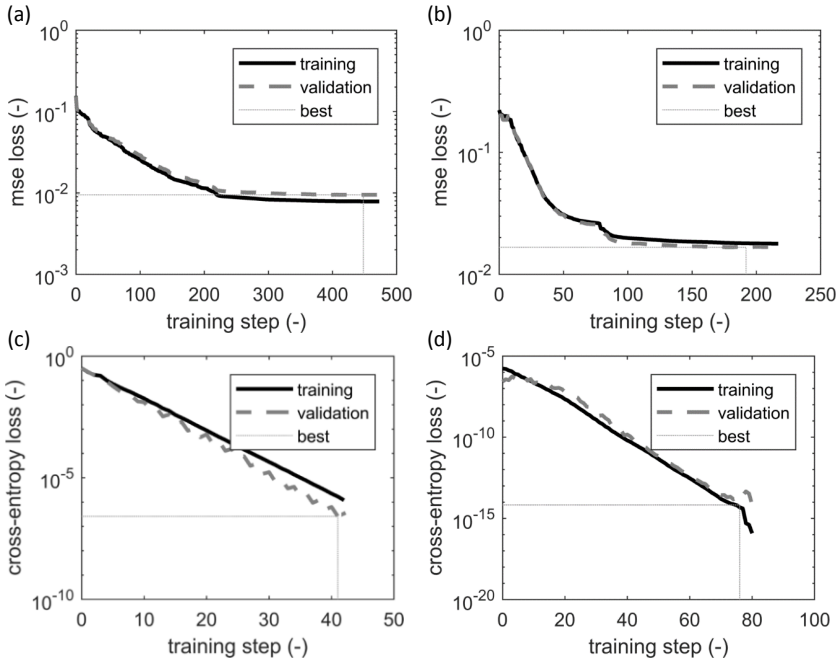

Figure 4.10: Learning curves: (a) the first autoencoder, (b) the second autoencoder, (c) the softmax layer, (d) the stacked deep learning network.

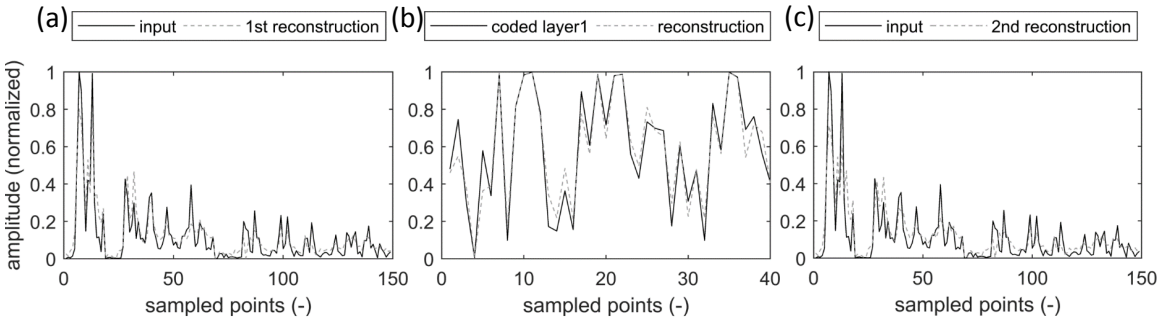

Figure 4.11: The input and output patterns of autoencoders: (a) the first autoencoder, (b) the second autoencoder, (c) reconstruction of the original input patterns after two layers of encoding and decoding.

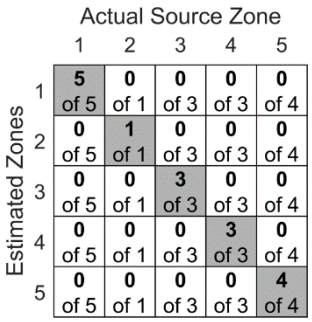

Figure 4.12: The confusion matrix of the stacked autoencoders in zonal localization of the first experimental scenario. Localization zone here is the closest rivet to the AE source.

Figure 4.13 shows the confusion matrix of the stacked autoencoders in the zonal localization of the second experimental scenario. Similar to the first experimental scenario, the stacked autoencoders were able to localize all AE sources. In this case, the testing set of the second experimental scenario included 42 AE sources that were randomly selected from a database of 416 simulated AE sources. It is important to note that the randomization used for the first and second experimental scenarios were different because each contained a different number of AE sources.

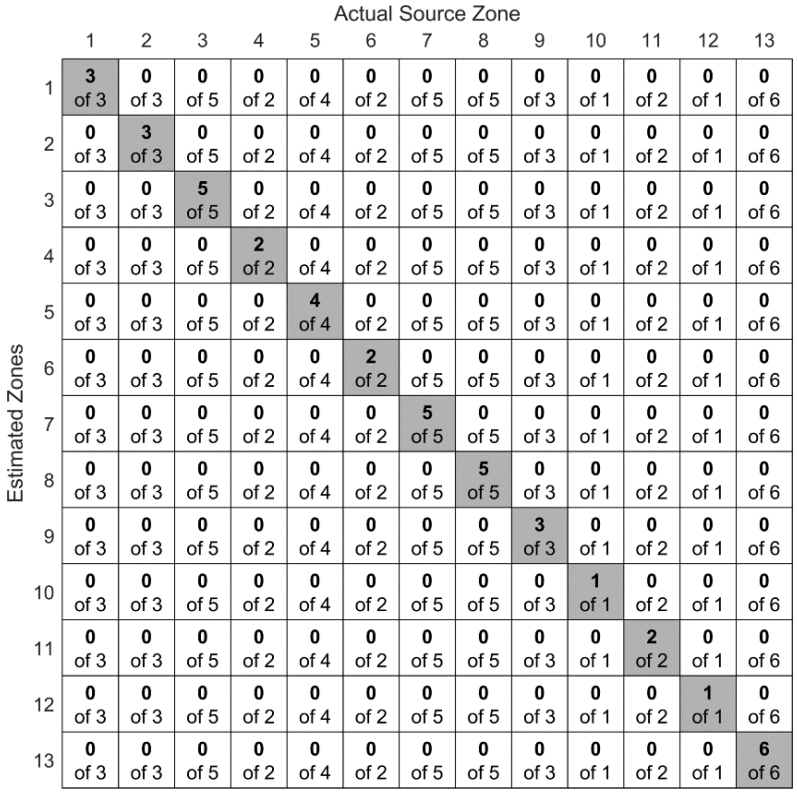

Figure 4.13: The confusion matrix of the stacked autoencoders in zonal localization of the second experimental scenario.

The time required to train the stacked autoencoders on a core-i5 processor was 28 seconds. However, it only takes less than 2 milliseconds for a trained network to localize a source.

4.6.2. Convolutional Neural Networks

Figure 4.14 visualizes the learning curves of the convolutional neural network. Since the network classifies the AE sources into five rivet locations, a cross-entropy loss was minimized. In particular, a gradient descent with momentum was used (Murphy 2012). In this study, the learning rate was 0.005 and the momentum contribution was 0.9. To avoid overfitting, in addition to a dropout layer, a weight decay regularization was used. Moreover, cross-validation was used to control the training process.

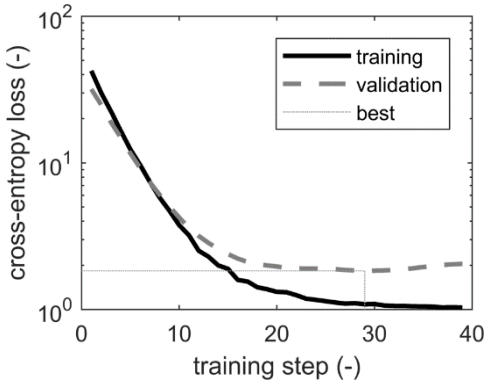

Figure 4.14: Learning curves of the convolutional neural network.

Figure 4.15(a) shows images that activate the final layer of the convolutional neural network the most. The five neurons in this layer correspond to the AE sources that occur near the five rivet connections. These images, which are generally called “deep dreams”, represent the inception of a convolutional neural network from the wavelet image of each rivet (Mordvintsev, Alexander Olah and Tyka 2015).

Figure 4.15(b) shows analytically calculated arrival time for edge-reflected late arrivals that appear in the coda of AE waveforms. Specifically, the arrival times are shown for different frequencies. The time and frequency ranges are the same as the ones used for the input images of the convolutional neural network. Since in plate-like structures AE sources excite guided ultrasonic waves (Lamb waves in particular) and the propagation velocities of these waves are a function of frequency, the arrival time is not the same across frequencies. To calculate late arrivals, the Multipath ray tracking algorithm (Ebrahimkhanlou et al. 2016b) was used to track the propagation paths of the Lamb waves from each rivet to the sensor (see Figure 4.16). Then, the dispersion curves of the first symmetric (S_0) and anti-symmetric (A_0) Lamb wave modes were used to convert the propagation distance of each path to its propagation time (i.e., its time of flight). It is assumed that the high-frequency content (425 kHz specifically) of the faster propagating

mode (i.e., S_0) triggers the AE system. Accordingly, the time of flights of the higher amplitude mode (i.e., A_0) were converted to the arrival times.

Comparing Figure 4.15(a-b) it could be seen that the convolutional neural network has learned the frequency-dependent reverberation patterns that appear in the coda of AE waveforms. Since the sensor was placed away from the lines of the symmetry of the plate, there is always a difference in the arrival time of the reflections that come to the sensor from different edges. Deep learning leverages such time differences and learns how to interpret them in terms of the location of AE sources. If one uses the first boundary of the plate (left, bottom, or top) that reflects the waves in each propagation path to divide the late arrivals into three groups, the arrival time of the three groups match with the high amplitude areas identified in Figure 4.15(a). From the first to last rivet, it takes more time for the first two groups to arrive at the sensor. In addition, the higher the rivet number, the later that the second group arrives than the first group. In contrast with the first two groups, the arrival time of the third group decreases from the first to last rivet.

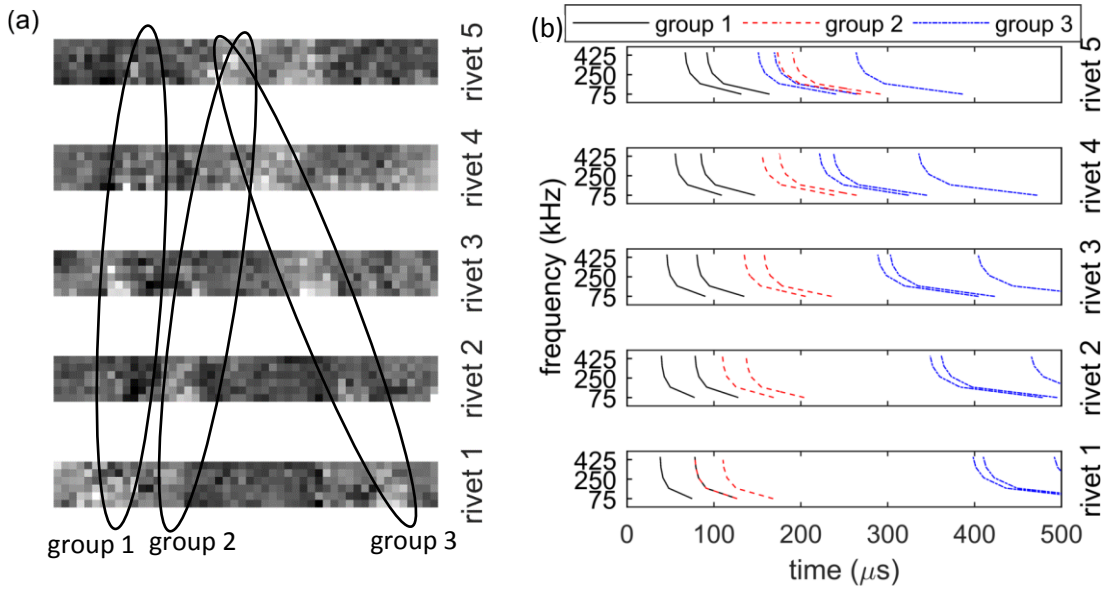

Figure 4.15: Three arrival groups: (a) deep dreams, (b) theoretical.

Figure 4.16 shows some possible propagation paths that connect the second and fourth rivets to the sensor. The paths are grouped based on their first reflecting boundary. For the sake of simplicity, only paths that interact with the left, bottom, or top boundaries are visualized. In addition, only paths with up to two reflections were considered. This is because the longer propagation paths, on which more reflections may occur, arrive after the 500-μs-long time window considered in this study.

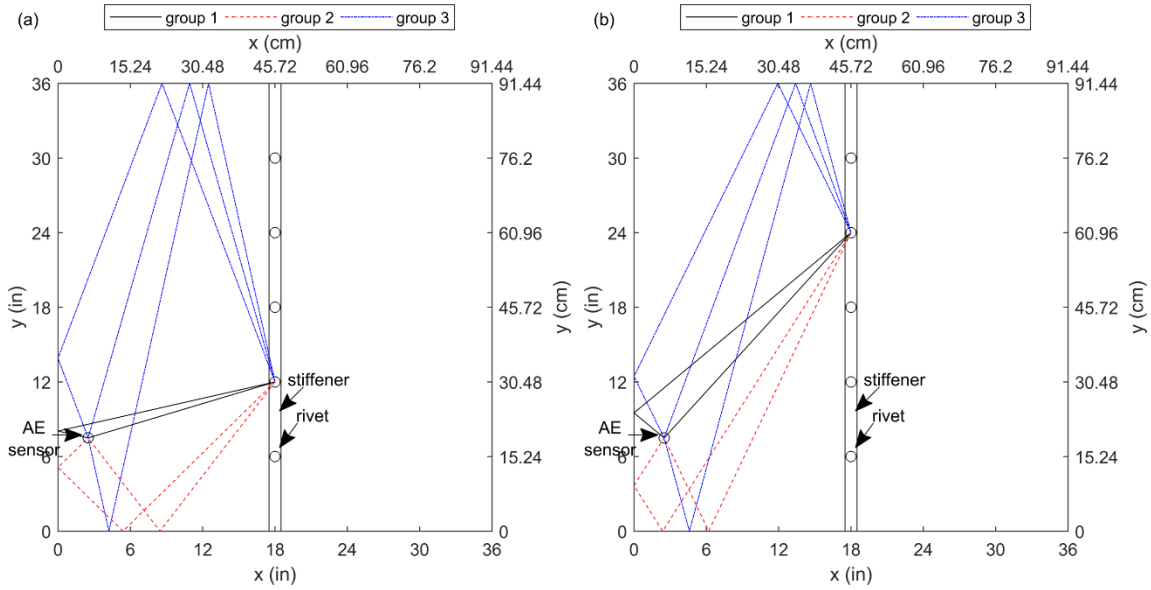

Figure 4.16: Propagation paths for the left side of the stiffener: (a) AE at the second rivet, (b) AE at the forth rivet.

Figure 4.17 shows the sensitivity map of the convolutional neural network. In this figure, the brighter the color map, the more sensitive the deep learning network to that particular part of the image. To produce these images a moving 10 × 4 occlusion window was used, and the average accuracy of the network was tested on each of the five rivets. As the figure shows, the network is the most sensitive to the arrival time of the three propagation groups identified in Figure 4.16. In addition, the network is more sensitive to the lower frequencies. These frequencies correspond to the most dispersive region of the higher-

amplitude Lamb wave mode that dominates the AE waveform (i.e., the first anti-symmetric mode). These observations further demonstrate that the convolutional neural network leverages the reverberation of AE waveforms as well as their dispersive behavior.

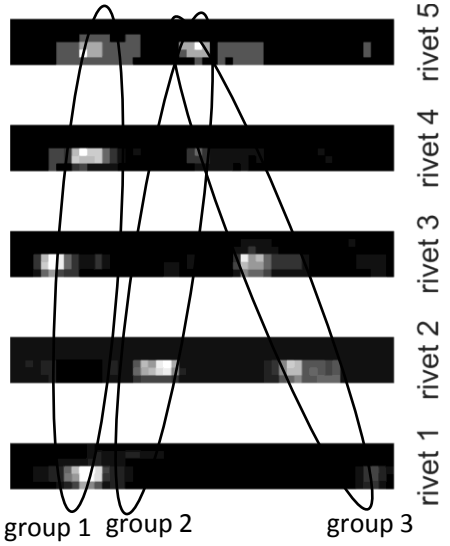

Figure 4.17: Sensitivity maps for a 10 × 4 occlusion; the three arrival groups are also indicated.

Figure 4.18 shows the confusion matrix of the convolutional neural network. The columns and rows of the matrix respectively represent the actual and estimated rivet numbers that are the closest to AE sources. The confusion matrix shows that the convolutional neural network, similar to the stacked autoencoders, successfully localized all AE sources in the testing dataset. For the sake of comparison, the same randomization was used for the two deep learning networks.

Figure 4.19 shows the confusion matrix of the convolutional neural network for the second experimental scenario. Overall, the accuracy of the network was 95.2%. Out of 42 pencil lead break tests, except for two, all simulated AE sources were correctly localized. One of the localization errors was for a source in zone nine (i.e., the top right surface of the plate)

that was inaccurately localized in zone five (i.e., the topmost rivet). In this case, the two zones are next to each other. The other error was for a source in zone eight (i.e., the top left surface of the plate), which was confused with a source in zone eleven (i.e., the bottom right edge of the plate).

The time required to train the convolutional neural network on a core-i5 processor was 23 seconds. However, it only takes less than 2 milliseconds for a trained network to localize a source.

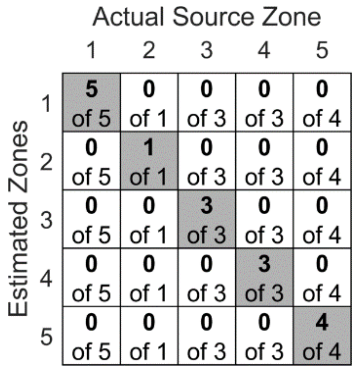

Figure 4.18: Confusion matrix of the convolutional neural network for the first experimental scenario. Localization zone here is the closest rivet to the AE source.

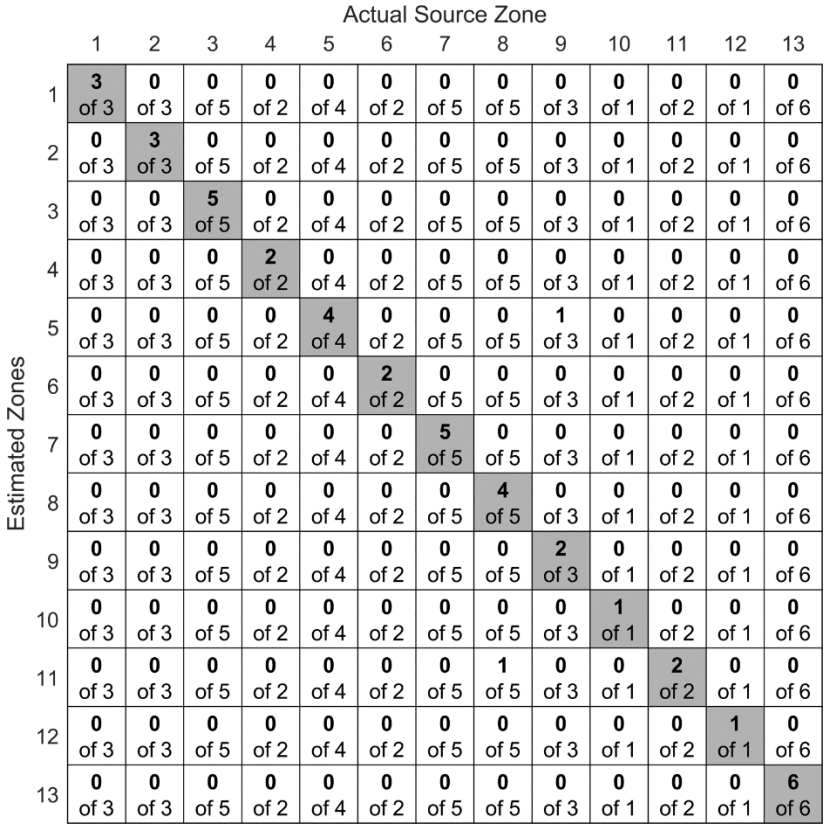

Figure 4.19: Confusion matrix of the convolutional neural network for the second experimental scenario.

4.7. DISCUSSION AND CONCLUSIONS

This paper used two deep learning approaches to localize AE sources within plates with geometric features, such as rivet-connected stiffeners. In particular, stacked autoencoders and convolutional neural networks were used. This paper leveraged the reflection and reverberation patterns of AE waveforms as well as their dispersive and multimodal characteristics to localize AE sources with only one sensor. To maximize the information attained by reflections, the sensor was attached to a corner of the plate, and symmetric locations, such as the center of the plate, were avoided. To train, validate, and test the deep learning networks, AE sources were experimentally simulated at the rivet connections of

an aluminum plate that had a stiffener. In particular, Hsu-Nielsen pencil lead break tests were used for the simulations. The results showed that both deep learning networks can learn how to map AE waveforms to their sources. These results demonstrate that the reverberation patterns of AE sources contain pertinent information to the location of their sources. Overall, the performance and flexibility of the two deep learning networks were comparable. While the stacked autoencoder achieved a slightly better performance (100% accuracy versus 95.2%), the convolutional neural network was more flexible in accepting more information-rich input in the frequency domain. In particular, six frequencies were used for the convolutional neural network compared to three in the stacked autoencoders.

This paper successfully performed zonal AE source localization. In particular, AE sources that may occur near rivet connections were localized. However, the current paper does not find the coordinates of AE sources. To overcome this limitation, future research may consider replacing the softmax layer of the deep learning networks with a regression layer and using a larger training data (Ebrahimkhanlou and Salamone 2018a). To generate a larger training data, future research should also focus on automating the process of simulating AE sources. For example, numerical simulations and/or robotic solutions could be investigated. While this study leveraged the broad-band frequency response of a PICO sensor to localize AE sources, in future, additional tests need to be performed to evaluate how a flat response would potentially improve source localization with deep learning. In addition, this paper used Hsu–Nielsen tests to simulate fatigue cracks. To verify the performance of the proposed deep learning approaches under actual states of stress, future researchers should perform more formal tests on real propagating cracks. Another option could be Hsu–Nielsen test performed at various depths inside rivet connections. Moreover, the scope of this paper was limited to embedded and permanently attached monitoring

systems that require one-time training before deployment. Since deep learning can also learn to generalize over the differences between different sensors and structures, future studies may investigate the idea of training a deep learning network on one structure and deploying the network on another similar structure.

4.8. ACKNOWLEDGMENTS

This work was supported by the Office of Naval Research under grant number N00014-17-1-2367, Program Director Dr. Ignacio Perez.

PAPER 5

A generalizable deep learning framework for localizing and characterizing acoustic emission sources in riveted metallic panels

Arvin Ebrahimkhanlou, Brennan Dubuc, and Salvatore Salamone[1]

5.1. SYNOPSIS

This paper introduces a deep learning-based framework to localize and characterize acoustic emission (AE) sources in plate-like structures that have complex geometric features, such as doublers and rivet connections. Specifically, stacked autoencoders are pre-trained and utilized in a two-step approach that first localizes AE sources and then characterizes them. To achieve these tasks with only one AE sensor, the paper leverages the reverberation patterns, multimodal characteristics, and dispersive behavior of AE waveforms. The considered waveforms include AE sources near rivet connections, on the surface of the plate-like structure, and on its edges. After identifying AE sources that occur near rivet connections, the proposed framework classifies them into four source-to-rivet distance categories. In addition, the paper investigates the sensitivity of localization results to the number of sensors. Moreover, the generalization of the deep learning approach is evaluated for typical scenarios in which the training and testing conditions are not identical. To train and test the performance of the proposed approach, Hsu-Nielsen pencil lead break tests were carried out on two identical aluminum panels with a riveted stiffener. The results

[1]This chapter has been submitted to an international journal to be considered for publication. The dissertation advisor has allowed the direct reproduction of chapter as a paper. The author of this dissertation was the lead investigator conduction the research. The author was responsible of developing theories, designing and conducting experiments, analyzing the results, and writing the journal paper. The coauthors contributed by providing editorial comments.

demonstrate the effectiveness of the deep learning-based framework as well as its potential for in-field implementation.

Keywords: acoustic emission, deep learning, edge reflection, reverberation patterns, plate-like structures, pattern recognition, stacked autoencoders, guided ultrasonic waves, machine learning, structural health monitoring

5.2. INTRODUCTION

Metallic panels are ubiquitous in structures such as ships and aircrafts. To minimize the maintenance costs and to increase the operation lifetime of these structures, researchers and practitioners are increasingly interested in improving current nondestructive evaluation (NDE) technologies or building advanced structural health monitoring (SHM) strategies. In particular, SHM strategies based on acoustic emissions (AE), have been widely used to localize (Kundu 2014) and characterize defects such as fatigue cracks in metallic panels (Holford et al. 2017; Zárate et al. 2012). In general, damage localization based on AE can be divided into two groups: (1) model-based and (2) model-free approaches. Model-based approaches usually require a detailed a priori knowledge of the structure (Dubuc et al. 2017). For example, the authors have recently developed a model-based approach that uses a combination of the reverberative, multimodal, and dispersive characteristics of AE waveforms to localize AE sources with only one sensor (Ebrahimkhanlou et al. 2016b; Ebrahimkhanlou and Salamone 2017d; e). Model-free algorithms, instead, require either more sensors or a set of training data to compensate for the lack of a priori knowledge. For instance, Sen and Kundu (2018) have recently proposed an approach that leverages the geometric shape of AE wavefronts to perform AE localization in anisotropic plates. The

approach requires at least nine sensors, and it can be used to localize AE sources occurring only within the convex area enclosed by the sensor clusters.

Data-driven approaches are another category of model-free approaches that rely on a set of training data. In contrast to the other abovementioned approaches that are usually limited to simple plates, data-driven approaches are applicable to any plate-like structure with no limitations on its geometry or features. This includes plate-like structures with rivet connected-stiffeners, which are the focus of this paper. Data-driven approaches use a set of training data to specialize in the task of AE source localization. In other words, these approaches implicitly *learn* the geometry and the material properties of the plate-like structure through the training data. Examples of this category include approaches based on neural networks (Sharif-Khodaei et al. 2012), delta-T mapping (Al-Jumaili et al. 2016b), and reciprocal time reversal (Ciampa and Meo 2011). Deep learning is a data-driven approach that can learn sophisticated input and output relationships. One main difference between deep networks and traditional shallow neural networks is that the input to deep networks can be signal itself, rather a few features extracted from it. In the context of AE, a deep network has the potential to take the entire length of waveforms as its input without relying on traditional AE features, such as time of arrival. Many traditional AE features were defined at the time that the AE technology emerged, when recording and analyzing the full length of AE waveforms was not possible. Therefore, the features were used in lieu of the waveforms. Deep learning, instead, has the potential to automatically identify representative patterns and features in AE waveforms. Compared to other data-driven techniques, one particular strength of deep learning lies in its high sensitivity to relevant details, and corresponding indifference to irrelevant aspects (Goodfellow et al. 2016; LeCun et al. 2015). Therefore, deep learning has the potential to identify and leverage

patterns in AE waveforms, which are not represented by traditional features. For example, to the best of the authors' knowledge, no AE feature represents the reverberation patterns of AE waveforms. However, the authors have recently used deep learning to learn the reverberation patterns of AE waveforms and leveraged them to classify the zone in which an AE source occurs (Ebrahimkhanlou and Salamone 2018b). In this way, they reduced the required number of sensors to only one and demonstrated that deep learning can leverage the reverberation patterns, dispersive behavior, and multimodal characteristics of AE waveforms. Nevertheless, they suggested that future studies should investigate *coordinate-based* source localization using deep learning. They also suggested evaluating the performance of such localization under a scenario in which the structure used for training is different from the one used for the performance evaluation.

In addition to localization, another open problem in AE-based SHM is characterizing the size of fatigue cracks. Although, in general, cumulative values of some AE features, including *counts* and *energy*, are correlated with the progression of fatigue (Holford et al. 2017), such values cannot classify fatigue cracks based on their lengths. As an alternative approach, accurate source localization should theoretically allow tracking the length of a progressive fatigue crack stemming from a rivet connection. However, this is constrained by the fact that most source localization algorithms cannot achieve an accuracy sufficient for crack sizing. In addition, the progression of the crack and its vicinity to rivets and stiffeners invalidates the "simple plate" assumptions used in most algorithms. Invalidating this assumption further reduces the accuracy of source localization algorithms, and hence their ability to track the length of cracks. As another alternative approach, Bhuiyan et al. (2017) recently showed a dependency between the frequency content of AE waveforms and the length of fatigue cracks. In particular, they used numerical simulations

and experiments to demonstrate that AE waveforms reflect back and forth between the two tips of a crack and thus create a resonance identifiable in the frequency domain. Deep learning could be an alternative approach to crack characterization. However, to the best of the authors' knowledge, little research exists on deep learning-based characterization of fatigue-related AE sources occurring in plate-like structures. This is despite the successful application of data-driven methods and particularly deep learning in other fields, including AE-based diagnosis and characterization of gearbox and bearing faults (He and He 2017; Jia et al. 2016; Li et al. 2016),

To localize and characterize AE sources with only one sensor, this paper proposes a deep learning-based framework that consists of three deep networks (see Figure 5.1). The idea is to 1) leverage the reverberation patterns of AE waveforms in the time domain to localize their sources, and 2) leverage the frequency content of AE waveforms to characterize fatigue-related AE sources. In particular, localization consists of zonal localization (deep network 1) and coordinate-based localization (deep network 2). First, zonal localization distinguishes AE sources that occur near rivets from those that occur elsewhere (e.g., on the surface or the edges of plate-like structures). In this paper, "near a rivet" means at or closer than 6.4 mm (1/4″) to the edge of the rivet. This definition defines a zone around each rivet, and another zone for the rest of the plate-like structure. After zonal localization, the coordinate-based localization (deep network 3) finds the coordinates of the sources that occur away from any of the rivets. As an initial step toward characterizing sources that occur near a rivet, the deep learning approach categorizes AE sources occurring near a rivet into multiple source-to-rivet distance categories. Since these AE sources usually stem from fatigue cracks, their distances to the edge of the rivets reveal the size of the cracks.

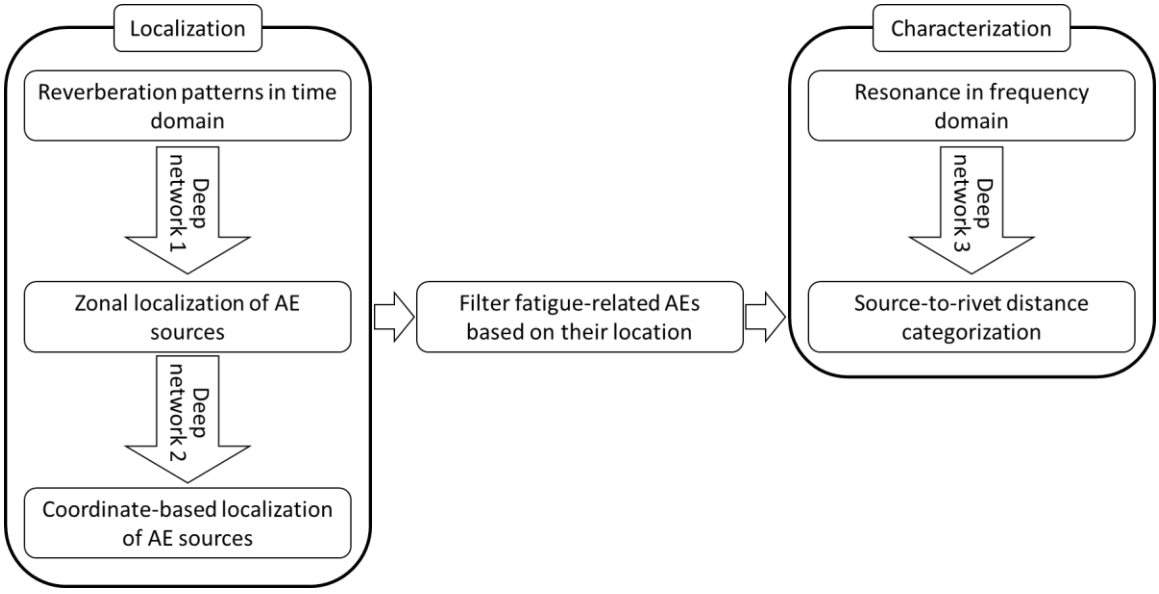

Figure 5.1. The proposed deep learning-based framework.

The overarching goal of the deep learning-based framework is to train the deep networks on one plate-like structure (or even a portion of it) and then deploy them on any replica of the structure. This eliminates the need to train a new deep network for every instance of the same plate-like structure, so too the need to re-train the deep network after a slight change in the structure. To this end, this paper also investigates such potential.

The subsequent sections of the paper are organized as follows. The next section briefly presents the theoretical aspects of source localization and characterization using deep learning. Afterward, the experimental setup is discussed. Finally, the results are presented and followed by concluding remarks.

5.3. DEEP LEARNING-BASED AE SOURCE LOCALIZATION AND CHARACTERIZATION

Several deep learning architectures have been introduced in the literature, including restricted Boltzmann machines, stacked autoencoders, convolutional neural networks, and long short-term memories (Goodfellow et al. 2016; LeCun et al. 2015). In particular, the

authors have recently compared stacked autoencoders and convolutional neural networks for zonal AE source localization (Ebrahimkhanlou and Salamone 2018b). Although most input-output architectures (e.g., stacked autoencoders and convolutional neural networks) could be adapted to AE source localization application, this paper focuses on stacked autoencoders because of their relative simplicity. Figure 5.2 shows a schematic view of this architecture. The input to the deep network (i.e. $\mathbf{x}$) is a one-dimensional vector extracted from the continuous wavelet transform of an AE waveform. The deep network consists of m encoder layers, which gradually compress (encode) the input and reduce its size. Depending on the task that the network performs (i.e. localization or characterization), the final layer of the network may respectively be a regression or a softmax layer. A regression layer maps the encoded features to the coordinates (i.e. *X*, *Y*) of the AE source, whereas a softmax layer classifies the AE source. For example, a softmax layer can classify AE sources based on their zonal location, source-to-rivet distance, etc. To manage the training process, in this architecture, the parameters of each layer are first pre-trained individually followed by fine-turning the entire network. The following subsections provide more details on the input, layers, and training process of the proposed deep learning architecture.

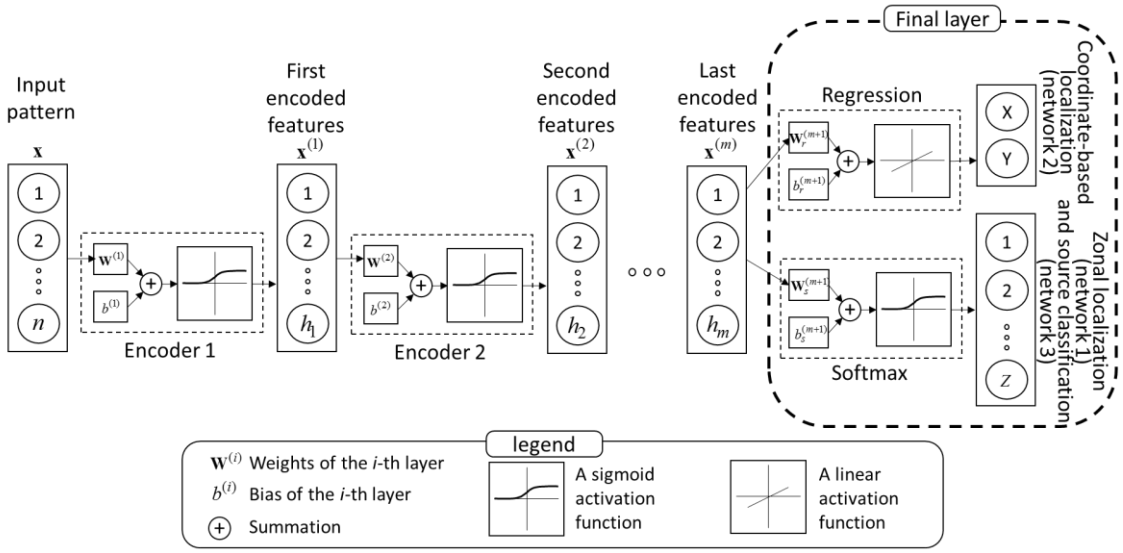

Figure 5.2. Stacked autoencoder architecture for deep learning; the last layer could be a regression layer or a softmax layer.

5.3.1. Input pattern

The following describes the process of constructing the input to the deep network. Let $r(t)$ be an AE waveform. In order to capture the multimodal and dispersive properties of the AE in addition to their reverberative characteristics, a time-frequency analysis based on a continuous wavelet transform is carried out (Mohammadi-Ghazi et al. 2018; Sarrafi et al. 2018b; a; Sarrafi and Mao 2018):

$$C(f,t)=\frac{1}{\sqrt{s(f)}}\int_{-\infty}^{+\infty} r(\tau)\Psi^{*}(\frac{\tau-t}{s(f)})\mathrm{d}\tau \tag{5.1}$$

In this equation, f and t are the frequency and translation parameters of the wavelet transform. In addition, $\Psi(t)$ and $s(f)$ are the mother wavelet and the non-dimensional scale parameter of the transform, respectively. In this notation, the star superscript indicates a complex conjugate. The complex Morlet mother wavelet is used as:

$$\Psi(t)=\frac{1}{\sqrt{\pi f_{\mathrm{b}}}}\exp(2\pi f_{\mathrm{c}} jt-\frac{t^{2}}{f_{\mathrm{b}}}) \tag{5.2}$$

where f_c (central frequency) and f_b (bandwidth) are two user-defined, non-dimensional parameters. In an earlier publication, the authors have shown that $f_b = 0.5$ and $f_c = 5$ are appropriate for AE in metallic panels (Ebrahimkhanlou et al. 2016b). The scale parameter is also defined as:

$$s(f) = \frac{f_c f_s}{f} \tag{5.3}$$

where f_s is the sampling frequency of $r(t)$. Figure 5.3 shows the normalized modulus of the complex wavelet transform coefficients for a typical AE waveform in a plate-like structure. The bright regions in the figure show that frequencies between 75kHz to 450 kHz contain the main energy of the waveform.

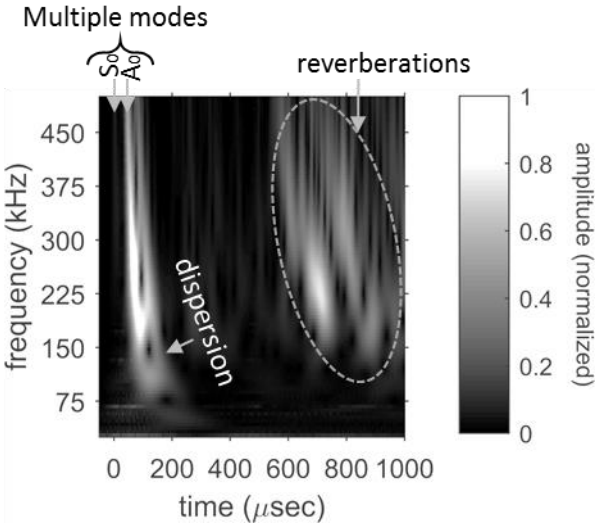

Figure 5.3. The complex wavelet transform of an AE waveform (the modulus of the wavelet coefficients).

Since the input to stacked autoencoders is a vector, this paper adapts the two-dimensional wavelet image to a one-dimensional input pattern (a vector). The first step to this process is to select the frequencies that have the highest amplitudes and best represent the AE waveforms. Figure 5.4 visualizes the process for a scenario in which six frequencies

are uniformly selected from the high-energy frequency range of the AE waveforms: 75, 150, 225, 300, 375, and 450 kHz. The next step is to calculate the modulus of the wavelet coefficients at these frequencies. Hereafter, wavelet coefficients at each frequency are referred to as a "wavelet slice" (see Figure 5.4(a)). In this paper, a 1-ms-long time window was used. The pre-trigger time, used to recover the AE waveform before the first threshold crossing, was set to -10 μs (negative time here means before threshold crossing). Next, each wavelet slice is re-sampled at the same frequency that they represent. For example, the wavelet slice at 75 kHz is resampled at the resampling frequency of 75 KHz. The reason for resampling is to eliminate redundant data points and make computations more efficient. The resampling also implicitly places more emphasis on higher frequencies. Although low frequencies contain essential information about the dispersive behavior, they have fewer wave packets than higher frequencies (see Figure 5.4(a)), and thus convey less information. Finally, the resampled wavelet slices are concatenated to construct the input pattern $\mathbf{x}$ (see Figure 5.4(b)).

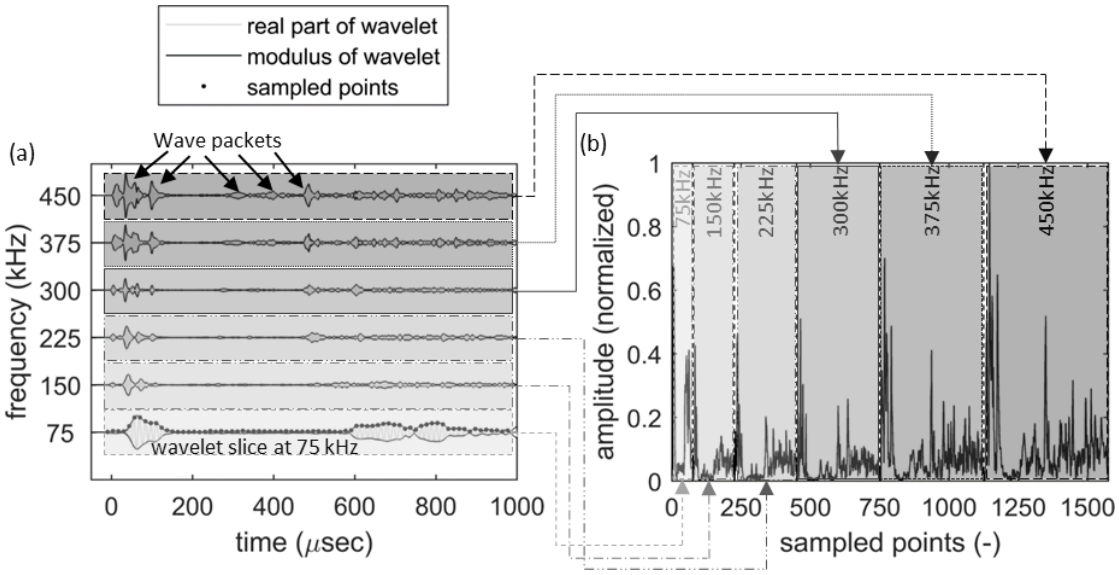

Figure 5.4. The wavelet transform of an AE waveform is resampled and concatenated to construct the input to the deep network: (a) the real part and modulus of the wavelet coefficients at six frequencies, for the sake of image clarity, only samples extracted from the lowest frequency (75 kHz) are indicated (b) the input to the deep network.

5.3.2. Autoencoders

Autoencoders aim at information compression. This goal is achieved by training them to compress their input to a code that after decompression reconstructs the input (Goodfellow et al. 2016). The compression and decompression tasks are called encoding and decoding, respectively. Figure 5.5 shows a typical autoencoder, which first transforms its input vector $\mathbf{x}^{(i-1)}$ into a smaller encoded vector $\mathbf{x}^{(i)}$, and then decodes it to a vector with the same size as $\mathbf{x}^{(i-1)}$. In the figure, the decoded vector is indicated as $\hat{\mathbf{x}}^{(i-1)}$. In this notation, the hat indicates *reconstruction*. The sizes of the input vector and the encoded vector are h_{i-1} and h_i, respectively, with $h_{i-1} > h_i$. By minimizing the difference between $\mathbf{x}^{(i-1)}$ and its reconstructed version $\hat{\mathbf{x}}^{(i-1)}$, autoencoders compress essential information from $\mathbf{x}^{(i-1)}$ in $\mathbf{x}^{(i)}$. Therefore, $\mathbf{x}^{(i)}$ implicitly contains *features* automatically extracted

from $\mathbf{x}^{(i-1)}$. To perform this task, autoencoders minimize a loss function, which is typically the mean squared error between $\mathbf{x}^{(i-1)}$ and $\hat{\mathbf{x}}^{(i-1)}$ (Vincent et al. 2008):

$$E = \frac{1}{N}\sum_{j=1}^{N} \| \mathbf{x}_j^{(i-1)} - \hat{\mathbf{x}}_j^{(i-1)} \|^2 \tag{5.4}$$

This process is commonly termed *training*. In Eq. (5.4), N is the number of AE waveforms used as training examples, and the subscript j indicates the j-th training waveform. Minimizing this loss function enables autoencoders to learn a nonlinear map that transforms $\mathbf{x}^{(i-1)}$ to $\mathbf{x}^{(i)}$:

$$\mathbf{x}^{(i)} = g(\mathbf{W}^{(i)}\mathbf{x}^{(i-1)} + b^{(i)}) \tag{5.5}$$

as well as another map that transforms $\mathbf{x}^{(i)}$ to $\hat{\mathbf{x}}^{(i-1)}$:

$$\hat{\mathbf{x}}^{(i-1)} = g(\mathbf{W}_{\mathrm{d}}^{(i)}\mathbf{x}^{(i)} + b_{\mathrm{d}}^{(i)}) \tag{5.6}$$

In these equations, $\mathbf{W}^{(i)}$, $b^{(i)}$ are the parameters (i.e. weights and biases) of the encoder, and $\mathbf{W}_{\mathrm{d}}^{(i)}$, and $b_{\mathrm{d}}^{(i)}$ are the parameters of the decoder, respectively. In addition, g is called the activation function. An established transition function is the sigmoid function (Bishop 2006), which is used in this paper:

$$g(\xi) = \frac{1}{1+e^{-\xi}} \tag{5.7}$$

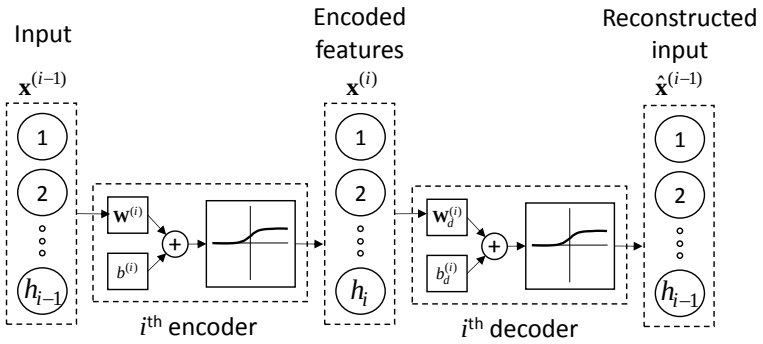

Figure 5.5. Example of an autoencoder.

Trained autoencoders are the building blocks of deep stacked autoencoders. In this deep learning architecture, each layer is trained to reconstruct the encoded features by the previous layer. Despite the name of stacked autoencoders, as Figure 5.2 shows, only the encoder portions of autoencoders are stacked together to construct the deep network.

5.3.3. Final layer of stacked autoencoders

The final layer of a deep network depends on the task that it performs. For a deep network that performs regression to find the coordinates of an AE source, the final layer is a "regression layer". On the other hand, the final layer of a deep network that performs classification to characterize an AE source or performs zonal localization is a "softmax layer". The next subsections provide more details on each of the regression and softmax layers.

5.3.3.1. Regression layer for localization of AE sources

A regression layer is a linear map between the encoded features by the last autoencoder layer $\mathbf{x}^{(m)}$ and the coordinates of AE sources. Let the actual coordinate of an AE source be $\mathbf{y}=[X,Y]^{\mathrm{T}}$. The regression layer estimates $\mathbf{y}$ as $\hat{\mathbf{y}}$. For a deep network that has m encoders:

$$\hat{\mathbf{y}} = \mathbf{W}_r^{(m+1)}\mathbf{x}^{(m)} + b_r^{(m+1)} \tag{5.8}$$

where $\mathbf{W}_r^{(m+1)}$, and $b_r^{(m+1)}$ are the weights and bias of the regression layer.

To train the weights and the bias of the regression layer, a loss function needs to be minimized. Given a set of training data containing input patterns $\mathbf{x}$ extracted from AE waveforms and their corresponding source coordinates $\mathbf{y}$, the loss function is defined as the mean squared error between $\mathbf{y}$ and $\hat{\mathbf{y}}$:

$$E_r = \frac{1}{N}\sum_{j=1}^{N} \| \mathbf{y}_j - \hat{\mathbf{y}}_j \|^2 \tag{5.9}$$

In this notation, N is the number of AE waveforms used for training, and the subscript j indicates the j-th training data point.

5.3.3.2. Softmax layer for characterizing AE sources

A softmax layer is a nonlinear map between the encoded features from the last autoencoder layer $\mathbf{x}^{(m)}$ and a finite number of classes that characterize AE sources. In this paper, AE sources that occur near rivet connections are divided into four source-to-rivet distance categories. Let $\mathbf{z}$ be a four-element vector with only one element equal to one and the three others zero. In this way, the element that is equal to one indicates the category to which an AE source belongs. The softmax layer estimates $\mathbf{z}$ with $\hat{\mathbf{z}}$, a vector whose elements sum to one, three of which are near zero. For a deep network that has m encoders:

$$\hat{\mathbf{z}} = g(\mathbf{W}_s^{(m+1)}\mathbf{x}^{(m)} + b_s^{(m+1)}) \tag{5.10}$$

where $\mathbf{W}_s^{(m+1)}$, and $b_s^{(m+1)}$ are the weights and the bias of the softmax layer. Mathematically, the softmax layer has a similar formulation to an encoder layer (see Eq. (5.5)) except that the activation function is different, which is typically a softmax function. For the i-th element of an arbitrary vector ξ, the softmax function is defined as:

$$g(\xi_i) = \frac{\exp(\xi_i)}{\sum_j \exp(\xi_j)} \tag{5.11}$$

Note that in this equation, the subscripts indicate the elements of the vector ξ. To train the weights and the bias of the softmax layer, usually a cross-entropy loss function is minimized (Bishop 2006). Given a set of training data containing input patterns $\mathbf{x}$ extracted from AE waveforms and their corresponding source category $\mathbf{z}$, the cross-entropy loss function is defined as:

$$E_s = -\sum_{j=1}^{N} (\mathbf{z}_j \cdot \ln \hat{\mathbf{z}}_j) \tag{5.12}$$

5.3.4. Training stacked autoencoders

Training is an optimization process that tunes the weights and biases of a deep network (Bishop 2006). The training of stacked autoencoders consists of two main stages: In the first stage, the weights and biases of each layer are trained individually. Hereafter, the weights and biases are collectively referred to as tunable parameters $\boldsymbol{\theta}$. To train the tunable parameters, a small random value is first assigned to each. Then, the parameters of each layer are iteratively updated until they minimize the corresponding loss function of the layer. It is important to note that, in this stage, the parameters of only one layer are updated at a time. However, in the second stage, the tunable parameters of the entire network are iteratively fine-tuned to minimize the loss function of the last layer (see Eq. (5.9) and Eq. (5.12)).

The training algorithm for both training stages is the same. Most deep learning architectures use the gradient descent algorithm to minimize their loss functions (Bishop 2006). This algorithm is a stochastic optimization algorithm that updates the tunable parameters $\boldsymbol{\theta}$ (i.e. weights and biases) of a deep network by taking small steps in the opposite direction of the gradient of the loss function $E(\boldsymbol{\theta})$:

$$\boldsymbol{\theta}_{(i+1)} = \boldsymbol{\theta}_{(i)} - \alpha \nabla E(\boldsymbol{\theta}_{(i)}) \tag{5.13}$$

In this equation, i is the iteration number, and α is the learning rate, which has a small positive value. To calculate the gradient of the loss function $\nabla E(\boldsymbol{\theta})$, stacked autoencoders use an algorithm named “backpropagation” (Bishop 2006). To improve the optimization, several numerical algorithms have been developed, including the scaled

conjugate gradient algorithm(Møller 1993), which this research uses in particular to solve Eq. (5.13).

5.3.4.1. Overfitting mitigation

Deep networks, like other machine learning algorithms, are susceptible to overfitting. Overfitting results in a deep network that only achieves a good performance on the training data and fails to perform well on unseen data. Several approaches exist in the literature to avoid overfitting, including cross-validation and regularization, which are both used in this paper (Bishop 2006; Goodfellow et al. 2016). In particular, cross-validation withholds a subset of the training dataset, named the *validation set*, from the training process. Then, it evaluates the loss function on the validation set and stops the training process when a minimum is reached. Additionally, regularization adds a new term to the loss function to prevent overfitting:

$$E'(\boldsymbol{\theta}) = E(\boldsymbol{\theta}) + \lambda \Omega(\boldsymbol{\theta}) \tag{5.14}$$

where $E'(\boldsymbol{\theta})$ is the regularized loss function, λ is the regularization coefficient, which usually has a small value (0.1 in this paper), and $\Omega(\boldsymbol{\theta})$ is the regularization term:

$$\Omega(\boldsymbol{\theta}) = \frac{1}{2}\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\theta} \tag{5.15}$$

5.4. EXPERIMENTS

To evaluate the performance of the proposed approach, Hsu-Nielsen pencil lead break tests (Hsu 1977) were conducted on two identical 6061-T6 aluminum panels with dimensions of 91.44 cm × 91.44 cm × 0.32cm (36″ × 36″ × 1/8″). A 6061-T6 aluminum doubler of dimensions of 91.44 cm × 2.54 cm × 0.32cm (36″ × 1″ × 1/8″) was riveted in the middle of each panel in order to add geometric complexity, as shown in Figure 5.6(f). Five rivets of 6.35 mm (1/4") diameter were used to connect the doubler to the plate. In

each Hsu-Nielsen test, the lead of a 2H-type, 0.5-mm (0.02″) mechanical pencil was 3 mm (0.12″) protruded and then broken on the surface and the edges of the panels at a 45-degree angle. To detect AEs, piezoelectric sensors (PICO sensor, Physical Acoustics) were attached to the panels with hot glue. Unless otherwise stated, the sensors were attached at the coordinates ($X = 6.35\,\text{cm}$, $Y = 19.05\,\text{cm}$) equivalent to ($X = 2.5''$, $Y = 2.5''$) measured from the lower left corner of the panels (see Figure 5.6(a) and Table 5.1). To amplify the waveforms by 40 dB, analog preamplifiers (2/4/6 switch selectable gain, Physical Acoustics) were used. Then, a data acquisition system (Micro Express, Mistras) digitized the waveforms at a sampling frequency of 5 MHz. The threshold and analog bandpath filter of the data acquisition system were 40 dB and 5 kHz-1000 kHz, respectively. Finally, the recorded waveforms were exported to MATLAB for post-processing.

Table 5.2 summarizes the experiments carried out on the two panels. Specifically, seven datasets were collected. The first dataset consists of 160AE waveforms generated by pencil lead breaks (PLB) carried out near each rivet connection. Specifically, 32 AE sources were generated near each rivet. As Figure 5.6(g) shows, these AE sources were generated at four sides of each rivet (i.e. right, left, top, bottom) in four source-to-rivet distances (i.e. at 0.8 mm, 1.6 mm, 3.2 mm, and 6.4 mm equivalent to 1/32", 1/16", 1/8", 1/4" measured from the edge of the rivet holes). Note each AE source was generated twice. The source-to-rivet distances were selected in such a way that correspond to the length of early stage to advanced cracks. The second dataset contains 3610 AE waveforms generated by PLBs on a grid with dimension of 5.08 cm × 5.08 cm (2" × 2") (see the white grid in Figure 5.6(b)). This dataset is used to generate training data for both the zonal and coordinate-based source localization approaches.

The grid includes AE sources on the surface and the edges of the plate. To account for the stochastic nature of AE waveforms, ten PLB sources were simulated per each point. The third dataset combines the first dataset with 160 data points randomly selected from the second dataset, in order to create training, validation and testing data for the zonal source localization. The selected data points from dataset II were located at least 6.4 mm (1/4") away from the rivets. The fourth dataset consists of AE waveforms generated by PLB tests carried out at points on a grid with dimensions of 11.43 cm × 11.43 cm (4.5" × 4.5"), as shown in Figure 5.6(b) (see black grid). This dataset represents the testing data for the coordinate-based source localization. To ensure proper evaluation of the deep learning approach and avoid overfitting to the training data, the training and testing grids were selected in such a way that causes minimum overlapping between the intersections of the two grids. In addition, note that the fourth dataset does not include any AE source at the coordinates (45.72 cm, 45.72 cm) equivalent to (18", 18"), which is the coordinates of rivet number three. In other words, despite the grid intersection shown at this coordinates, no PLB sources were simulated there for the fourth dataset.

This paper also investigates the generalization of the deep learning approach over different specimens as well as its performance when the number of sensors increases. To investigate the generalization, the fifth dataset was created to include AE sources similar to those in dataset IV, except that this time pencil lead break tests were carried out on the second panel (see Figure 5.6(c)). To investigate the sensitivity of localization results to the number of sensors, four AE sensors were attached to one of the panels (see Table 5.1 for the coordinates of the sensors) and pencil lead break tests were carried out on its lower left quarter of the panel (see Figure 5.6(d)). In particular, the sixth dataset contains 1000 AE waveforms collected from sources at points on a grid with dimensions of 5.08 cm × 5.08 cm

(2" × 2"), as shown in Figure 5.6(d) (see the white grid). The seventh dataset was created to test the deep network trained by using the sixth dataset. Specifically, the seventh dataset contains 16 AE waveforms collected from PLB tests at points on a grid with the dimensions of 11.43 cm × 11.43 cm (4.5" × 4.5") (see the black grid in Figure 5.6(d)). Finally, the paper investigates the generalization of source characterization over the training sensor. To this end, the eighth dataset contains AE sources similar to those in dataset one except that the coordinates of the sensor were different: (19.05 cm, 6.35 cm) equivalent to (7.5″, 2.5″) (see Figure 5.6(e)). This sensor is numbered as "sensor 5" in the figure as well as Table 5.1.

Table 5.1. Coordinates of AE AE sensors.

	Coordinates	
Sensor	S.I. units	U.S. units
1	(6.35 cm, 19.05 cm)	(2.5", 7.5")
2	(6.35 cm, 72.39 cm)	(2.5", 28.5")
3	(85.09 cm, 19.05 cm)	(33.5", 7.5")
4	(85.09 cm, 72.39 cm)	(33.5", 28.5")
5	(19.05 cm, 6.35 cm)	(7.5", 2.5")
1*	(6.35 cm, 19.05 cm)	(2.5", 7.5")

* Attached to specimen 2

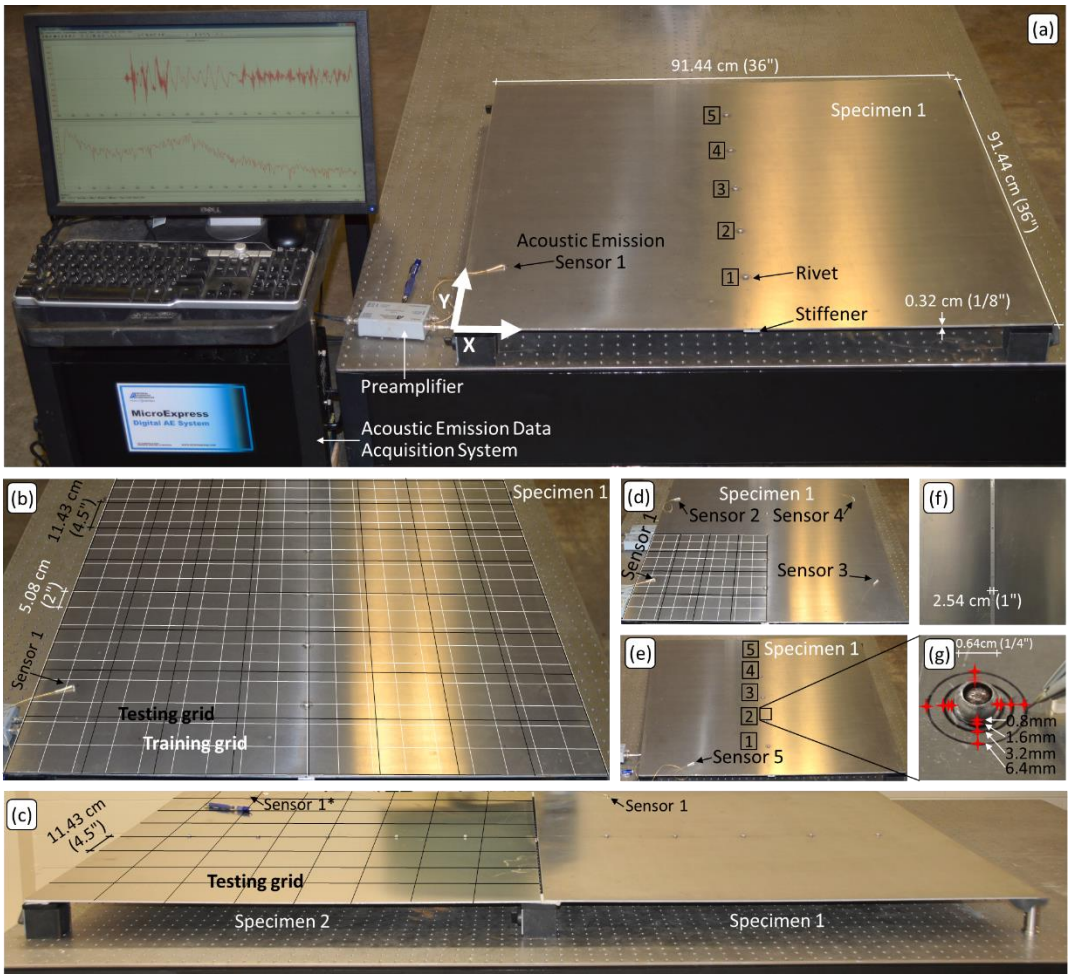

Figure 5.6. Experimental setup: (a) overview (sources at rivets), (b) sources at grid intersections (separate grids for training and testing), (c) testing on specimen 2 (sources at grid intersections), (d) the effect of the number of sensors (sources at grid intersections), (e) back view of the plate (f) training with one sensor and testing with another (sources at rivets), (g) 32 sources at 16 locations around each rivet.

Table 5.2. Pencil lead break datasets

Dataset	Specimen	Experi-mental setup	Number of sensors	Number of pencil lead break tests	Number of pencil lead breaks at the same coordinates	Description
I	1	a	1	160	2	AE sources in the vicinities of the rivets: at four sides and the distances of 0.8 mm (1/32″), 1.6 mm, (1/16″), 3.2 mm (1/8″), and 6.4 mm (1/4″) (see Figure 5.6(g))
II	1	b	1	3610	10	AE sources at the intersections of the white grid shown in Figure 5.6(b). The grid spacing was 5.08 cm (2″)
III	1	a,b	1	320	1	Dataset I plus 160 randomly selected data points from dataset II. The selected data points from dataset II were away from rivets
IV	1	b	1	48	1	AE sources at the intersections of the black grid shown in Figure 5.6(b). The grid spacing was 11.43 cm (4.5″)
V	2	g	1	48	1	Similar to dataset II except that the specimen was different (see Figure 5.6(c)).
VI	1	c	4	1000	10	AE sources at the intersections of the white grid shown in Figure 5.6(d). The grid spacing was 5.08 cm (2″)
VII	1	c	4	16	1	AE sources at the intersections of the black grid shown in Figure 5.6(d). The grid spacing was 11.43 cm (4.5″)
VIII	1	d	1	64	2	Similar to dataset I except for the location of the sensor (see sensor 5 in Table 5.1) and that the tests were performed only for the first two rivets (see Figure 5.6(e)).

5.5. RESULTS AND DISCUSSION

Table 5.3 summarizes the seven deep learning-based studies carried out in this paper. The first study performs zonal source localization. It specifically determines whether AE sources have occurred near any rivet and, if so, it identifies the closest rivet to the source. For the sources identified as away from the rivets, the second study finds their coordinates. Next, the third study investigates the performance of the coordinate-based source localization on another identical structure. In addition, the fourth study evaluates

the sensitivity of the coordinate-based source localization to the number of AE sensors. Then, the fifth study categorizes AE sources that occur near rivet connections into four source-to-rivet distance categories. Finally, the sixth and seventh studies investigate the accuracy of source categorization for scenarios in which a different rivet or a different sensor is used for training the deep network, respectively.

Table 5.4 lists all three deep networks used in this paper. The first network take wavelet coefficients at three frequencies to perform zonal source localization. The second network takes such coefficients at six frequencies to find the coordinates of AE sources. Finally, the last deep network takes wavelet coefficients at three frequencies to characterize AE sources that occur near rivets into four source-to-rivet distance categories. The last layer of the first and the third network is a softmax layer, whereas the last layer of the second network is a regression layer. The table also reports the computation time required to train each network measured on a core-i5 PC. The computation time for the second network is significantly longer than the other two networks. The reasons that contribute to this difference include: 1) the higher number of input frequencies that also increases the number of data points in the input layer (i.e. first layer), 2) the higher number of layers, 3) the higher number of nodes in each layer, and 4) the use of a regression layer as the last layer.

Table 5.3. Source localization and characterization studies

Study	Specimen	Test setup	Num. of sensors	Deep network	Training dataset	Validation dataset	Testing dataset	Description
1	1	a,b	1	1	80% of dataset III (256 data points)	10% of dataset III (32 data points)	10% of dataset III (32 data points)	Zonal localization: whether sources occur near rivets, if so, which rivet
2	1	b	1	2	90% of dataset II (3249 data points)	10% of dataset II (361 data points)	100% of dataset IV (100 data points)	Coordinate-based localization
3	1,2	b,g	1	2	90% of dataset II (3249 data points)	10% of dataset II (361 data points)	100% of dataset V (49 data points)	Extension to study 2: Training on specimen 1, testing specimen 2 (structure generalization)
4	1	c	1-4	2	90% of dataset VI (900 data points)	10% of dataset VI (100 data points)	100% of dataset VII (16 data points)	Extension to study 2: Investigating the effect of the number of sensors
5	1	a	1	3	80% of dataset I (128 data points)	10% of dataset I (16 data points)	10% of dataset I (16 data points)	Categorizing AE sources based on their distance from rivets
6	1	a	4	3	72.5% of dataset I (116 data points collected at four of the rivets)	7.5% of dataset I (12 data points collected at four of the rivets)	20% of dataset I (all 32 data points at the remaining rivet)	Extension to study 5: investigates training on four of the rivets and testing on the remaining one (rivet generalization)
7	1	a,d	1	3	90% of dataset I (144 data points)	10% of dataset I (16 data points)	100% of dataset VIII (64 data points)	Extension to study 5: investigating the potential for training with one sensor and testing with another (generalization over sensor and its location)

Table 5.4. Deep stacked autoencoders

Deep network number	Task	Frequencies (kHz)	Number of nodes in each layer					Final layer	Training time
			1st layer	2nd layer	3rd layer	4th layer	5th layer		
1	Zonal localization	75, 200, 325	600	40	15	6	N.A.	Softmax	1 min
2	Coordinates localization	75, 150, 225, 300, 375, 450	1575[a], 3150[b], 4725[c], 6300[d]	200	50	10	2	Regression	2 hours
3	Characterization	55, 155, 190	360	40	15	4	N.A.	Softmax	1 min

[a] single sensor, [b] two sensors, [c] three sensors, [d] four sensors

5.5.1. Training autoencoders

The first step to train stacked autoencoders is to train each autoencoder individually. For brevity, only the training curves of one of the deep networks are shown here. In particular, Figure 5.7 shows the mean squared errors between the input and output of each autoencoder layer in the second deep network. Note that the vertical axes have different scales and the reported errors include an L_2 regularization term (see Eq. (5.14)). To avoid overfitting, each training was stopped when a global minimum was achieved on a validation set. The figure shows the global minimums labeled as "best".

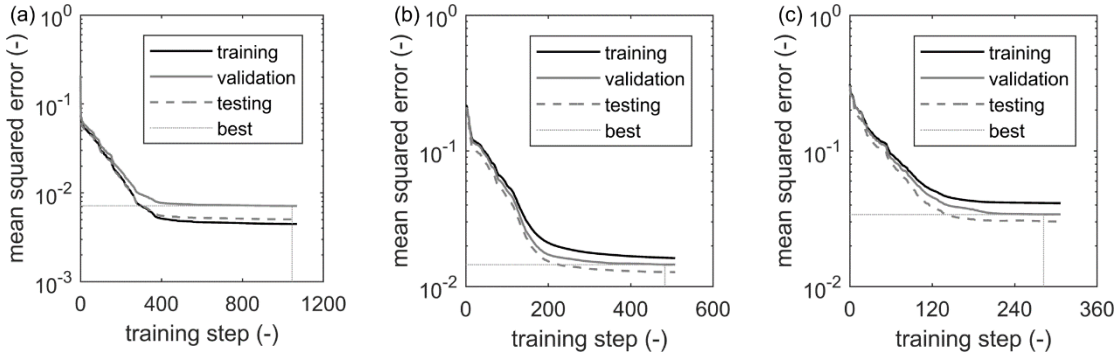

Figure 5.7. Learning curves for the autoencoders of deep network 2, the alphabetic labels represent the three autoencoder layers of the network (different vertical and horizontal scales).

Figure 5.8 compares the input and output of each autoencoder used for coordinate-based source localization (i.e. deep network 2). For brevity, this figure also shows the results obtained for one of the deep networks. Specifically, the figure shows that for every autoencoder, the output reconstructs the input. Since the input to each autoencoder is the encoded patterns by the previous autoencoder layer, the stacked autoencoders gradually reduces the size of the input pattern. As the figure shows, the size of the input pattern to the first layer is 1575, which is encoded to 200, and then 50 features.

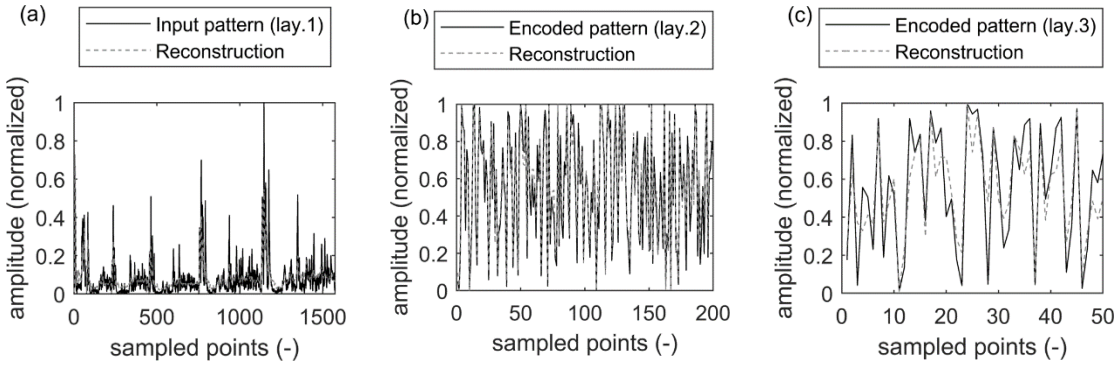

Figure 5.8. Comparison of the autoencoders' input patterns and their reconstruction, deep network 2, the alphabetic labels represent the three autoencoder layers of the network.

5.5.2. Zonal source localization

The first deep learning study of this paper is zonal source localization. Zonal source localization determines whether the AE source has occurred near any rivet connection and if so, it identifies the closest rivet connection to it. To this end, the first deep network was used (see Table 5.4). Since such determination is a classification task, the final layer of this deep network was a softmax layer (see Figure 5.2). To train and evaluate the performance of this deep network, dataset three was used. Specifically, the training, validation, and testing sets respectively included 256, 32, and 32 data points randomly selected without replacement from the dataset (see Table 5.3). As Table 5.2 shows, dataset three includes

pencil lead break sources near each rivet as well as sources on the surface or at the edges of the specimen. In particular, the specimen was divided into six zones: the first five zones being near the five rivets, respectively (see Figure 5.6(a)), and the sixth zone being anywhere else. In this paper, being near a rivet is defined as being in a distance less than or equal to 6.4 mm (1/4″) from the edge of rivets.

Figure 5.9 shows the confusion matrix of zonal source localization. A confusion matrix is a table that is often used to describe classification performance. The more number on the diagonal on this matrix the better the performance (less classification confusion). As the figure shows, the deep network correctly localized all 32 sources in the testing set. Since the testing set was randomly selected, the number of the sources in each zone was not equal.

Estimated Zone \ Actual Source Zone	Rivet 1	Rivet 2	Rivet 3	Rivet 4	Rivet 5	Elsewhere
Rivet 1	**4** of 4	**0** of 4	**0** of 2	**0** of 5	**0** of 2	**0** of 15
Rivet 2	**0** of 4	**4** of 4	**0** of 2	**0** of 5	**0** of 2	**0** of 15
Rivet 3	**0** of 4	**0** of 4	**2** of 2	**0** of 5	**0** of 2	**0** of 15
Rivet 4	**0** of 4	**0** of 4	**0** of 2	**5** of 5	**0** of 2	**0** of 15
Rivet 5	**0** of 4	**0** of 4	**0** of 2	**0** of 5	**2** of 2	**0** of 15
Elsewhere	**0** of 4	**0** of 4	**0** of 2	**0** of 5	**0** of 2	**15** of 15

Figure 5.9. Confusion matrix for zonal source localization on dataset III (study 1).

5.5.3. Coordinate-based source localization

The second and third deep learning studies aim at finding the coordinates of the AE sources that the zonal source localization (see section 5.5.2) classifies as “elsewhere”. Such

sources occur on either the surface or the edge of the plate-like structure (see Figure 5.10). To perform coordinate-based source localization, the second deep network was used (see Table 5.4). In particular, dataset II was used to train and validate this network (see Table 5.3). The difference between the second and third study is in the datasets used to test their performance. In particular, the second study used dataset IV, which was collected on the same structure as the one used to collect dataset II, and the third study uses dataset V, which was collected from another, albeit identically fabricated, plate-like structure. In both cases, the coordinates of AE sources used for training were different from those used for testing.

Figure 5.10(a-b) show coordinate-based source localization results for the second and third deep learning studies, respectively. In the former, the deep network was trained and tested on specimen 1, whereas in the latter, the trained network on specimen 1 was tested on specimen 2. Specifically, the mean squared localization error was 4.44 cm (1.75″) for Figure 5.10(a) and 9.32 cm (3.67″) for Figure 5.10(b). A comparison between the two figures demonstrates the generalization of the deep learning approach over a new plate-like structure. Despite the increase in error, these results suggest that a trained deep network on one plate-like structure can localize AE sources on another identical plate-like structure with reasonable accuracy.

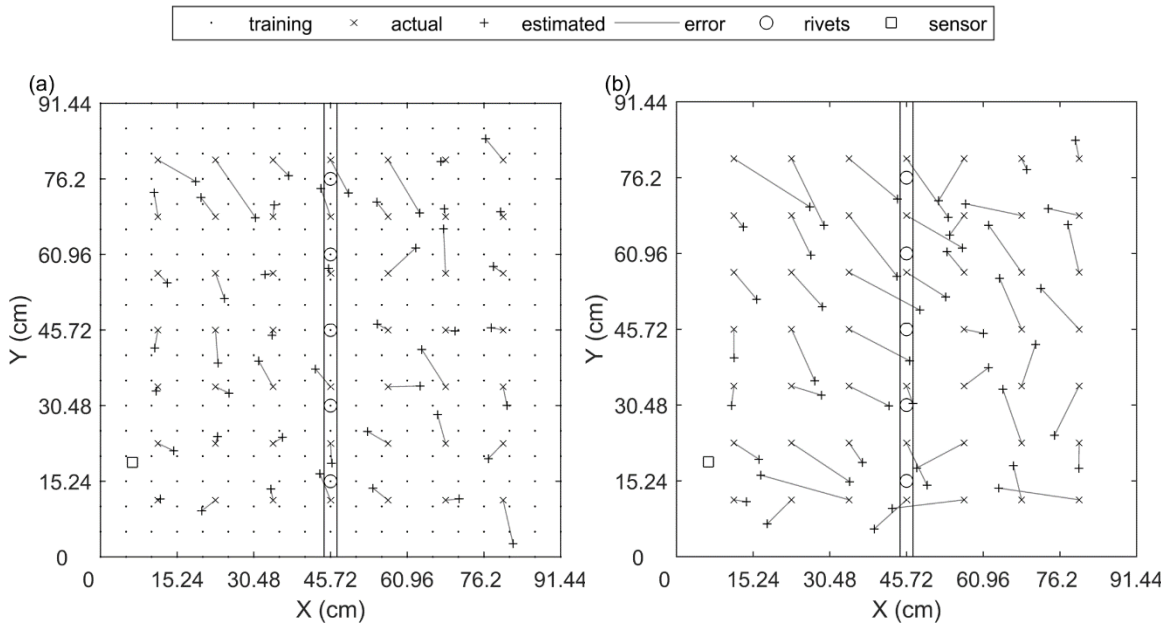

Figure 5.10. Localization results for AE sources located anywhere on the plate but away from rivets: (a) training and testing on specimen 1 (study 2), (b) training on specimen 1 and testing on specimen 2 (study 3).

To understand the distribution of coordinate-based localization error, Figure 5.11 shows the histogram of the error for the second and third deep learning studies. In particular, the distribution of the error was bell-shaped when the network was trained and tested on the same specimen, whereas it was relatively uniform when the same network was tested on a second specimen.

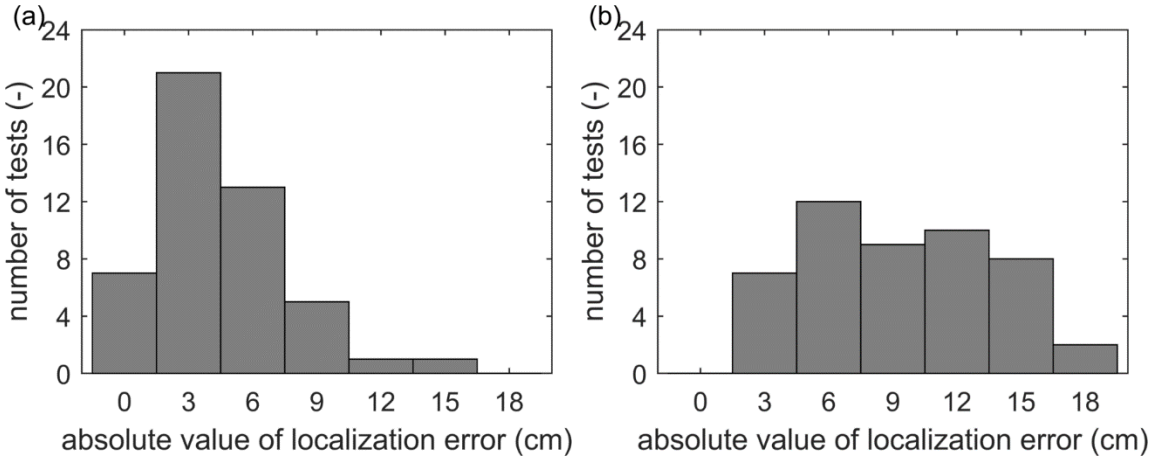

Figure 5.11. Histogram of localization error (absolute value): (a) training and testing on specimen 1 (study 2), (b) training on specimen 1 and testing on specimen 2 (study 3).

Figure 5.12 investigates the directionality dependence of the coordinate-based localization error for the second and third deep learning study. Specifically, each error vector is decomposed into two directions: 1) source-to-sensor direction (hereafter referred to as radial direction), and 2) normal to the source-so-sensor direction (hereafter referred to as tangential direction). For both studies, the error was larger in the tangential direction than the radial direction. Such results are consistent with the results published earlier by the authors showing a similar observation when an analytical model is used to perform single-sensor AE source localization in a simple plate (Ebrahimkhanlou and Salamone 2017d).

To investigate the dependence of coordinate-based localization error on the source-to-sensor distance, Figure 5.13 plots the two parameters with respect to each other and calculates a trendline for them. For the sake of consistency with the previous figures, results obtained from the second and third deep learning studies are presented side-by-side. Despite the low R-squared value of the trendlines, in both cases, the error tends to slightly increase as a function of source-to-sensor distance. However, such an increase was less

evident for the third study in which the trained deep network on specimen 1 was tested on specimen 2.

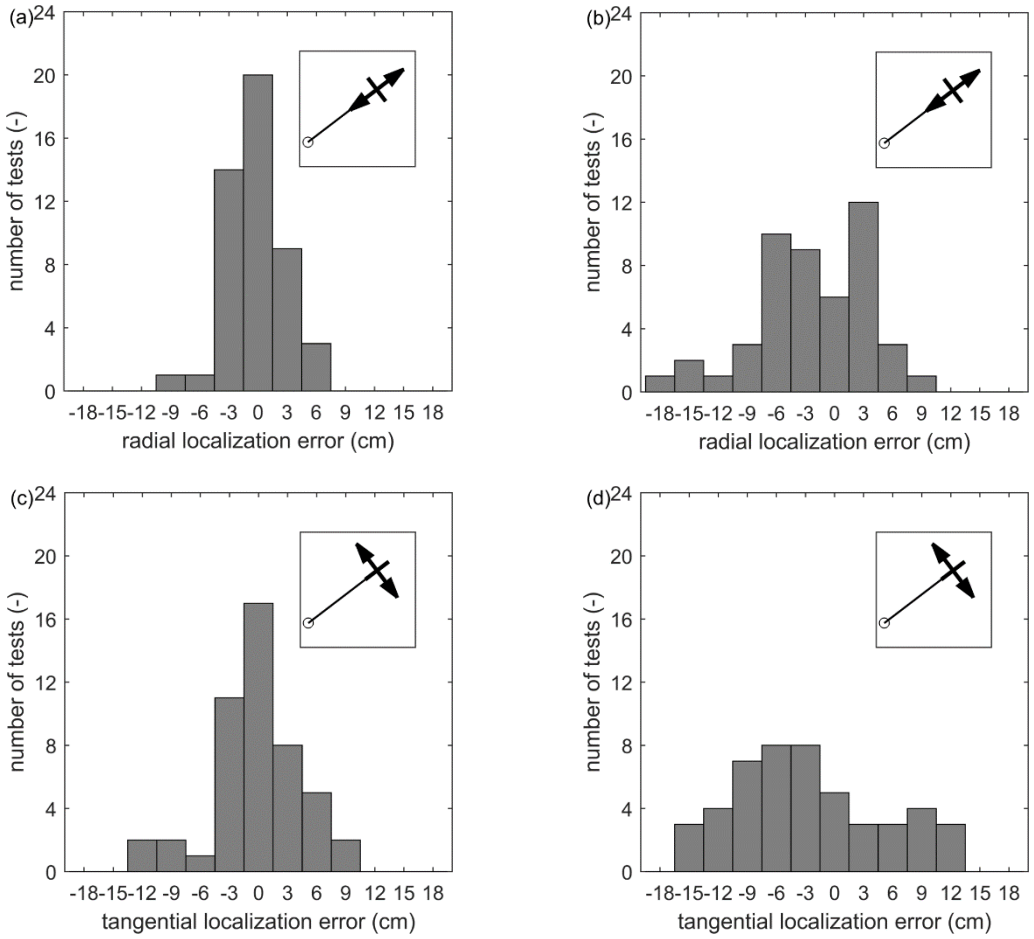

Figure 5.12. Histogram of localization error: (a) study 2, training and testing on specimen 1, radial direction (b) study 3, training on specimen 1 and testing on specimen 2, radial direction, (c) study 2, training and testing on specimen 1, tangential direction, (d) study 3, training on specimen 1 and testing on specimen 2, tangential direction.

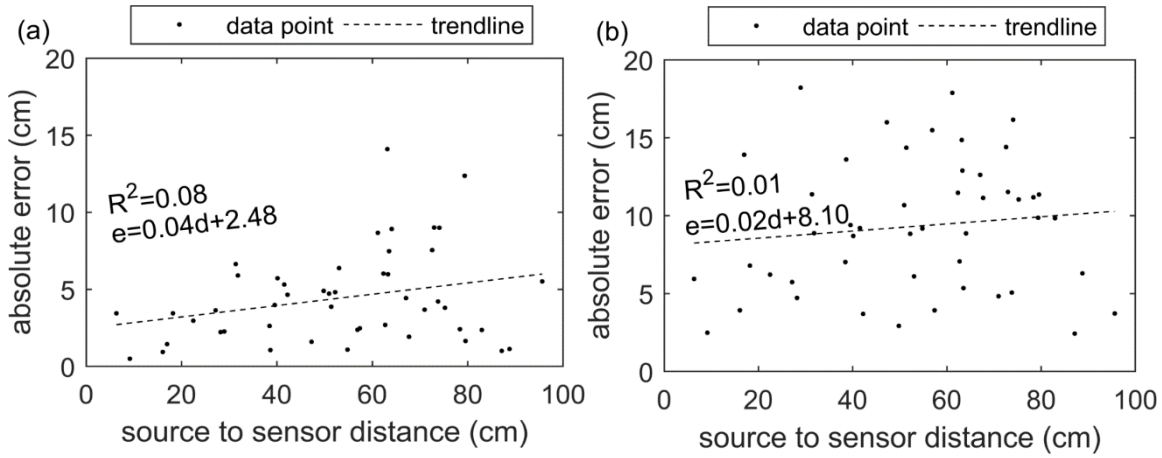

Figure 5.13. Localization error as a function of source-to-sensor distance: (a) study 2, (b) study 3.

5.5.4. Sensitivity analysis to the number of sensors

The fourth deep learning study investigates the performance of the coordinate-based localization algorithm as the number of AE sensors increases. To this end, AE waveforms in dataset VI and VII were respectively used to train and test deep network 3. Figure 5.14 shows typical examples of such waveforms simulated on the lower left corner of specimen 1 (see Figure 5.6(d)). The figure shows that the time of threshold crossing is not the same for all the sensors. To ensure that the first threshold crossing always occurs for sensor number one, pencil lead beak sources were simulated only on the lower left corner of the specimen. In this way, sensor 1 always triggered the data acquisition and the time stamp of the waveforms were independent of sensors 2, 3, and 4. Therefore, the same dataset could have been used to compare the effect of the number of sensors by including in or excluding from the localization algorithm the data obtained from sensors 2, 3, and 4.

Adding more sensors provides the deep learning-based source localization algorithm with additional data that can increase its accuracy in two ways: 1) each sensor provides more information on the dispersive, reverberative, and multimodal characteristics

of AE waveforms. 2) Having more than two sensors enables the deep network to leverage the time-difference of arrival to each sensor. To input the additional data to the deep learning algorithm, the wavelet patterns of the participating sensors were concatenated. Table 5.4 shows the number of input data points to the first layer of deep network 2.

Figure 5.15 compares results obtained using one to four sensors. In this figure, subplots (a-d) present results obtained from sensor 1, 1 and 2, 1-3, and 1-4, respectively (see Table 5.1 as well as Figure 5.6(d) for the location and numbering of the sensors). As Table 5.3 shows, the comparison is based on dataset VI and VII (see Table 5.2 for the definition of each dataset). Since the datasets contain pencil lead break sources only on the lower left corner of the specimen, Figure 5.15(b-d) show only this area of the specimen. The results show that adding more sensors can significantly reduce the localization error. In particular, Figure 5.15(e) shows that the mean squared error dropped from 4.4 cm to 1.3 cm when four sensors were used instead of one sensor. In other words, quadrupling the sensors decreased the error by a factor of three.

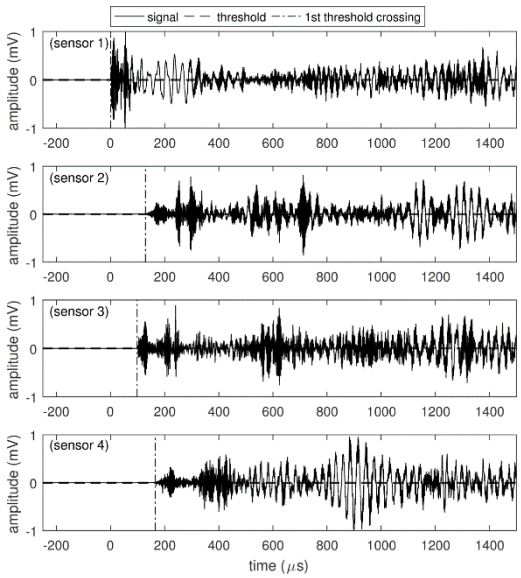

Figure 5.14. Sample AE signals collected by four sensors (dataset VI).

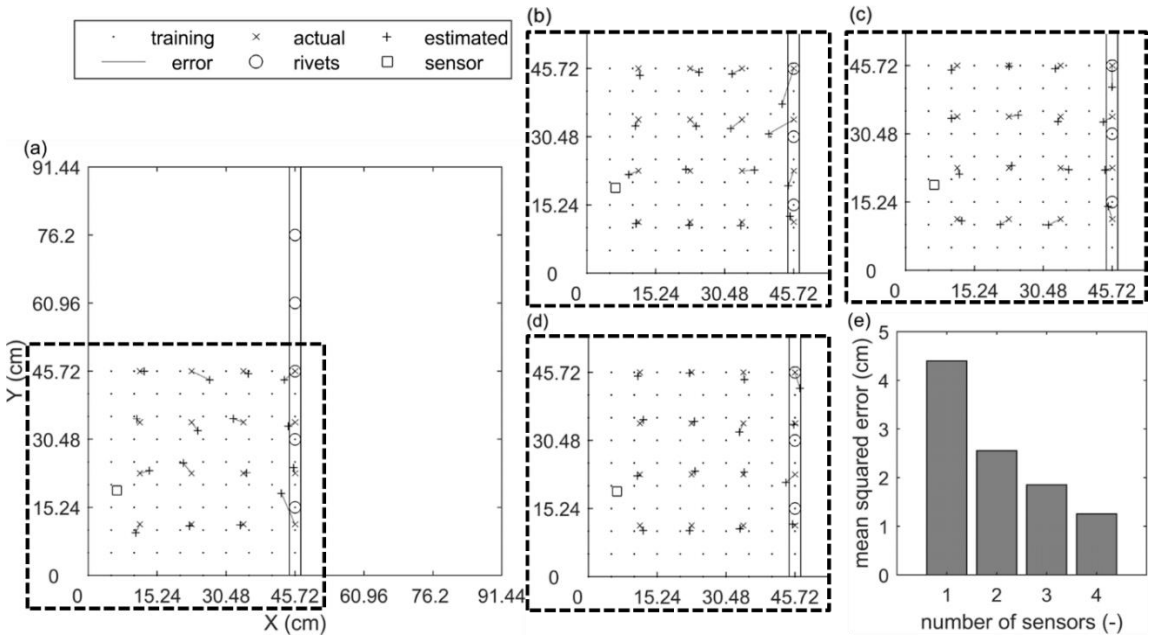

Figure 5.15. Sensitivity of coordinate-based localization results to the number of sensors (study 4): (a) one sensor, (b) two sensors (1 and 2), (c) three sensors (1-3), (d) four sensors (1-4), (e) comparison.

5.5.5. Characterizing AE sources

The fifth deep learning study characterizes AE sources that occur near rivets into four source-to-rivet distance categories: 1) 0.8 mm (1/32″), 2) 1.6 mm, (1/16″), 3) 3.2 mm (1/8″), and 4) 6.4 mm (1/4″). These are the sources that the zonal source localization identifies them near one of the rivets (i.e. in one of the rivet zones, see section 5.5.2). Since the tips of propagating fatigue cracks that stem from rivets are AE sources, the overarching goal is to infer the size of such cracks. To this end, dataset I is used to train, validate, and test the performance of deep network 3. This deep network is very similar to deep network 1, except for its output and input (see Table 5.4). The output of the network is one

of the abovementioned four source-to-rivet distance categories. The input patterns, however, were constructed from wavelet slices at frequencies that, on average, were most sensitive to the source-to-rivet distances. To identify such frequencies, Figure 5.16 shows the mean frequency spectrum for each source-to-rivet distance category as well as the range of variations. It could be seen that, at 55 kHz, 155 kHz, and 190 kHz, there is a distinctive difference between the frequency contents of the four source-to-rivet distance categories. Since deep learning can leverage such differences to discriminate the four source-to-rivet distance categories, wavelet slices at these three frequencies were used to construct the input patterns.

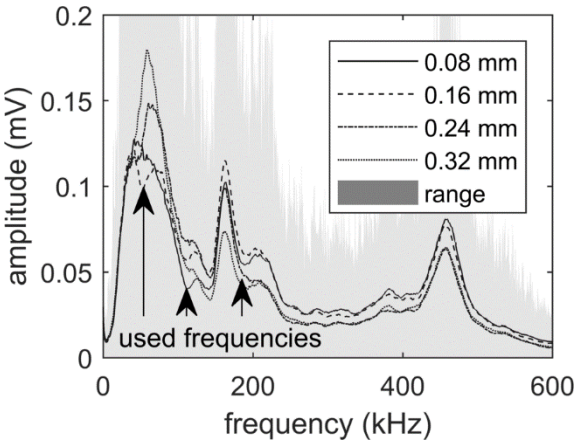

Figure 5.16. Fourier transform of AE waveforms: mean values for four source-to-rivet distance categories are plotted, the shaded area is the range of variations, and the arrows indicate the frequencies at which the plots disagree the most.

The overall accuracy of AE source characterization was 75.6%. Due to the small size of the dataset, the reported accuracy is an average of a ten-fold cross-validation. Figure 5.17 shows the breakdown of the results for four source-to-rivet distance categories. The horizontal axis is the actual distance, and the vertical axis is the estimated one. For each of the actual source-to-rivet distance categories on the horizontal axis, the circles and numbers indicate what percentage was characterized as one of the four estimated source-

to-rivet distances. For example, 75.0% of AE sources that were simulated at a 0.08 mm (1/32″) distance from the rivets were correctly characterized, and the remaining 25.0% were incorrectly estimated to have a source-to-rivet distance of 1.6 mm (1/16″). The figure also shows that the erroneous estimations were mainly misclassified in the most immediate categories. For instance, none of the AE sources that were simulated at a 0.16 mm (1/16″) distance from the edge of rivets were estimated in the 0.64 mm (1/4″) category. In this case, the erroneous estimations were 2.7% in the 0.32 mm (1/8″) category and 8.1% in the 0.16 mm (1/16″) category.

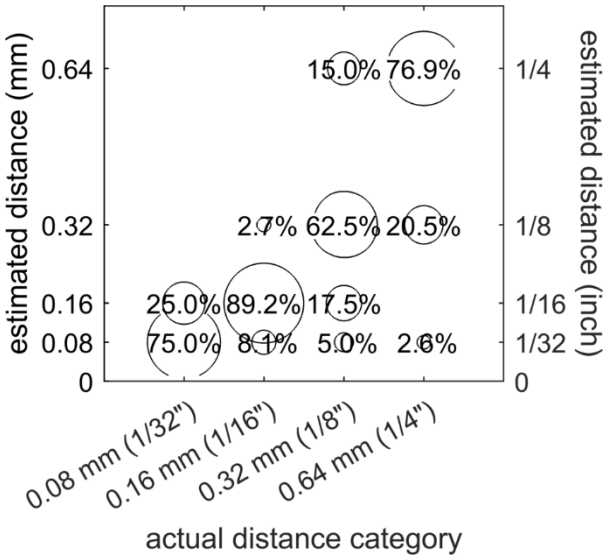

Figure 5.17. Categorizing AE sources that occur near rivets based on their source-to-rivet distances (study 5).

Since plate-like structures usually have a large number of rivet connections, it is not practical to train a deep learning algorithm for every individual rivet. Instead, it would be more desirable to train the network on a subset of the rivets and deploy it for all of them. In particular, Figure 5.18(a) shows results for the sixth study (i.e. source characterization) in which the training and validation sets were randomly selected from a portion of AE

sources in dataset I that were simulated near four out of the five rivets. Accordingly, the testing set of the sixth study consisted of AE sources in dataset I that were simulated near the remaining rivet (see Table 5.3). Hereafter, the rivet used for testing is referred to as the holdout rivet. To make the results independent of the holdout rivet, deep network 3was trained and tested ten times, rotating the holdout rivet. Then, the results were averaged. Overall, 66.3% of AE sources were characterized correctly. In addition, Figure 5.18(a), similar to Figure 5.17, shows that the majority of misclassified AE sources were categorized in the next immediate source-to-rivet distance categories. Such results prove that the deep learning approach can generalize source-to-rivet distance categorization over rivets for which it has never been trained.

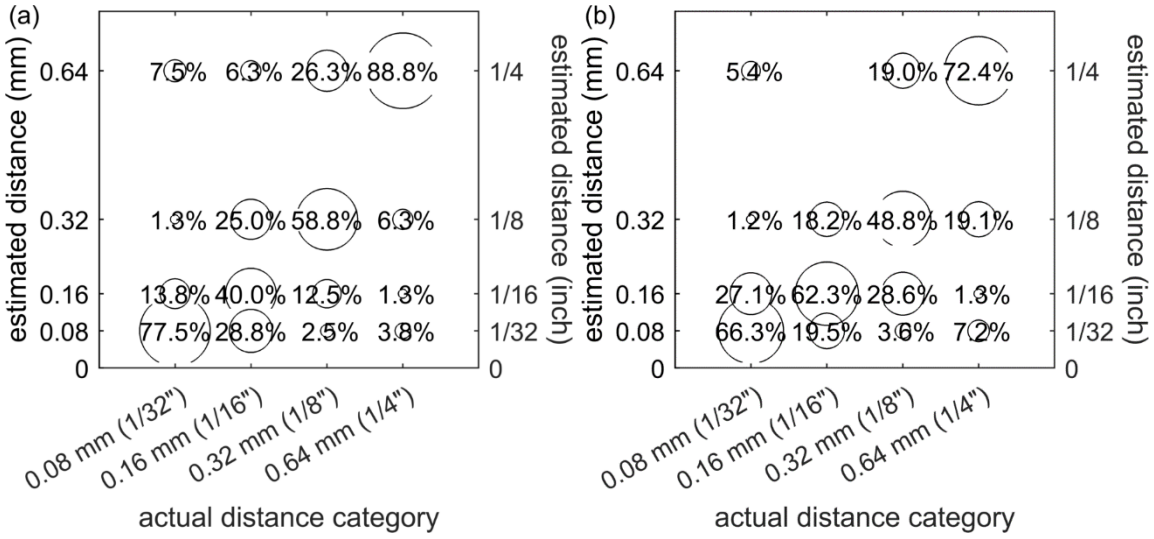

Figure 5.18. Categorizing AE sources that occur near rivets based on their distance from the edge of rivets: (a) training on four of the rivets and testing on the fifth one (generalization over rivets, study 6), (b) training on sensor 1 and testing on sensor 5 (generalization over sensor, study 7).

The seventh deep learning study investigates the performance of AE source characterization under another scenario in which there is a fundamental but realistic

difference between the training and testing data. In this case, the sensors used to train and test deep network 3 were attached to two different locations. In particular, the training data was dataset I, which was collected by sensor 1 attached to the coordinates (6.35 cm, 19.05 cm) equivalent to (2.5″, 7.5″), and the testing data was dataset VIII, which was collected by sensor 5 attached to the coordinates (19.05 cm, 6.35 cm) equivalent to (7.5″, 2.5″). Figure 5.6(a and e) show the location of sensors one and five, respectively. Although the sensors were of the same make and model, they were not strictly identical. Note also that the sensor-to-structure bond of no two sensors is identical. Overall, the accuracy of AE source characterization was 62.2%. To compensate for the size of the dataset, the reported numbers here are averages of a ten-fold cross-validation. Figure 5.18(b) shows the breakdown of the characterization accuracy. Similar to Figure 5.17 and Figure 5.18(a), the majority of misclassified AE sources were categorized in the next immediate source-to-rivet distance categories. The results support the independence of source-to-rivet distance categorization from the location of the AE sensor as well as its generalization over the individual sensor variations. In other words, the deep network does not need to be re-trained every time the sensor is changed, and the new sensor does not have to be attached at exactly the same coordinates as the previous one.

5.6. SUMMARY AND CONCLUSIONS

This paper introduced a deep learning-based framework for localizing and characterizing AE sources in complex, riveted metallic panels. The framework uses deep stacked autoencoders to perform a zonal source localization, which distinguishes whether the AE source is near one of the rivets or not. Then, it finds the coordinates of the sources that are not near any rivet. Finally, it characterizes fatigue-related AE sources that stem

from rivet connections into four source-to-rivet distance categories representing early stage to critical cracks. The framework collectively leverages the reverberation patterns, multimodal characteristics, and dispersive behavior of AE waveforms to reduce the number of sensors to only one. To validate the proposed framework, experiments were carried out on two aluminum panels with riveted doublers. The results showed the accuracy of both zonal and coordinates-based source localization. In particular, the coordinate-based source localization had no blind zones and could localize AE sources anywhere on the specimen. However, it showed a slight directional dependency by having less error in the source-to-sensor direction. It was also observed that the coordinate-based localization tends to have higher accuracy in localizing AE sources that were closer to the sensor. However, the low R-squared value of a trendline relating source-to-sensor distances and the errors did not permit a definite conclusion. In addition, the paper investigated the sensitivity of the coordinate-based localization results to the number of sensors. The results showed increasing the number of sensors from one to four reduced the localization error from the mean squared average of 4.4 cm to 1.3 cm. In other words, the errors were reduced by a factor of three. Such reduction could be attributed to leveraging the time difference of arrivals that occurs between the sensors in addition to the extra information that each sensor provides. Moreover, this paper considered scenarios in which the deep network is tested on a plate-like structure that is not exactly the same as the one used to train the deep network. The results showed that the deep learning-based framework could generalize over the structure and the sensor. Such results suggest that deep learning eliminates the need to train new deep networks for every identical copy of a plate-like structure. In terms of characterizing AE sources, the paper also investigated two other aspects of generalization. First, since plate-like structures usually have a large number of rivet connections, it is not

practical to train a deep learning algorithm for every individual rivet. Therefore, the deep network was trained on a subset of the rivets, but it was used to characterize all near-rivet sources. The paper also showed that the deep network does not need to be re-trained in the case of sensor replacement, and the new sensor need not be attached at exactly the same coordinates.

Despite the novel results presented in this paper, some limitations may be noted. Any data-driven approach, including deep learning, requires training data. Although the presented generalization results alleviate this limitation, the manual process of simulating AE source using pencil lead break tests is time-consuming and may not perfectly simulate real fatigue cracks. Therefore, future research should carry out fatigue tests and investigate the use of robotic arms or numerical simulations (Farhangdoust et al. 2017; Hamstad et al. 2001, 2003) to automate the process of generating training data. In addition, this paper did not consider environmental effects, operational noises, and the effect of connection to the supports and nearby panels should be accounted in future studies. Although the sensor used here had a relatively wide-band response, in future, other types of AE sensors, such as true wide-band sensors or MEMS (Kabir et al. 2015b, 2018a; b), may be considered.

5.7. ACKNOWLEDGMENTS

This work was supported by the Office of Naval Research [grant number N00014-17-1-2367], Program Director Dr. Ignacio Perez. The authors also express gratitude to the American Society for Nondestructive Testing (ASNT) for the ASNT Fellowship Award granted to the first author.

SUMMARY AND CONCLUSION

6.1. SUMMARY

The research presented in this dissertation aimed at advanced pattern recognition techniques for guided ultrasonic wave-based structural health monitoring (SHM) of metallic panels that are ubiquitous in civil, naval, and aerospace structures. The motivation behind this study was aging of such structures that are being used beyond their initial design life, and in some cases, are bearing more loads than they were initially designed for. Among the wealth of SHM techniques for metallic panels, techniques based on guided ultrasonic waves are among the most promising ones. However, one major challenge in guided wave-based monitoring of such panels is the large number of wave reflections from geometric features of the panels, such as edges, stiffeners, and rivet connections. Neglecting these large number of reflections, which are usually referred to as reverberations, results in false positive (i.e. detecting none-existing defects), undermines the effectiveness of the SHM system, and hinders it in-field implementation. Instead, the main goal of this research was to leverage the reverberation of guided ultrasonic waves in conjunction with the dispersive and multimodal characteristics of these waves to reduce the number of transducers required for SHM of metallic panels. To implement this new paradigm, an analytical and a data-driven approach were taken and two damage assessment modes were considered: active ultrasonic and passive acoustic emission (AE). However, more weight was given to the passive mode. In particular, uncertainties in localizing AEs were quantified, and AE sources were characterized. In all cases, experiments were carried out to evaluate the performance of the reverberation-based paradigm. The results were promising and showed the effectiveness of the new paradigm, especially, in reducing the number of transducers. In summary, the specific objectives of this study included the following:

1. Developing an analytical model to simulate the reverberation patterns of guided ultrasonic waves.
2. Leveraging the reverberation patterns simulated by the analytical model for active damage localization.
3. Comparing the results obtained from the reverberation-based active damage localization with existing algorithms, including Delay-and-Sum (DS) and Minimum Variance (MV).
4. Leveraging the reverberation patterns simulated by the analytical model for passive AE source localization.
5. Statistically analyzing errors involved in reverberation-based passive AE source localization.
6. Probabilistically quantifying random and systematic uncertainties in localizing AE sources.
7. Implementing a deep learning approach for reverberation-based AE source localization in metallic panels with geometric features, such as rivet-connected stiffeners.
8. Comparing the performance of two deep learning architectures (i.e. convolutional neural networks and stacked autoencoders) for AE source localization.
9. Statistically analyzing errors involved in coordinate-based localization of AE sources using deep learning.
10. Identifying AE sources that occur near rivet connections and characterizing them based on their source-to-rivet distances.
11. Quantifying the performance of the deep learning-based AE source localization when the training and testing conditions are not identical.

12. Analyzing the sensitivity of the deep learning-based AE source localization (coordinate based localization) to the number of AE sensors.

6.2. CONCLUSIONS

Based on the work performed and presented in papers 1-5, the following primary conclusions can be made:

Regarding active ultrasonic imaging:

- The reverberation-based image was able to localize a single defect with only three transducers.
- The approach was also able to produce an image of new defects in a metallic panel with pre-existing defects.
- Compared to Delay-and-Sum (DS) and Minimum Variance (MV) algorithms, which require more transducers and are applicable only to the convex area covered by transducers, the reverberation-based ultrasonic image was able to localize defects anywhere on a metallic plate.
- The computation time for producing a reverberation-based ultrasonic image was within a few seconds.

Regarding passive AE source localization using analytically simulated reverberation patterns:

- Reverberation-based AE source localization approach had no blind zones and was able to localize AE sources anywhere within a simple plate with only one sensor.
- The average and maximum localization errors were 2.8 cm and 8.2 cm, respectively.
- The localization error was less in the source-to-sensor direction than the direction normal to it.

- The computation time for calculating the location of each AE source was a few seconds.
- The effect of random and systematic uncertainties in localizing AE sources was quantified and presented as confidence contours.

Regarding passive AE source localization and characterization using deep learning:

- After dividing the metallic panel into predefined zones, the deep learning-based, AE source localization was able to find the zone within which an AE source had occurred. In particular, the deep learning approach was able to distinguish whether AE sources had occurred on the surface, at the edges, or next to the rivet connections of a metallic panel. In addition, the results showed that this approach could narrow down further and, for example, detect the closest rivet to the source.
- Two deep learning architectures, specifically, convolutional neural networks and stacked autoencoders, were compared and achieved nearly the same accuracy in zonal localization of AE sources. While the stacked autoencoder achieved a slightly better performance, the convolutional neural network was more flexible in terms of the format of its inputs.
- After analyzing the patterns leaned by deep learning, two techniques, namely deep dreams and occlusion, demonstrated that convolutional neural networks learn the *reverberation patterns* of AE waveforms.
- Single-sensor AE source localization had no blind zone and could localize AE sources anywhere on the surface, at the edges, or near rivet connections of a metallic panel.
- Localizing an AE source with deep learning only took a few milliseconds.

- Stacked autoencoders found the coordinates of AE sources in a 91.44 cm × 91.44 cm × 0.32 cm metallic panel with an average error of 4.4 cm (single sensor).
- The localization errors of the coordinate-based localization were smaller in the source-to-sensor direction than in the direction normal to it.
- The coordinate-based localization had higher accuracy in localizing AE sources that were closer to the sensor.
- In addition, the paper investigated the sensitivity of the coordinate-based localization results to the number of sensors. The results showed that including more AE sensors enabled the deep learning approach to leverage the time difference of arrivals that occurs between the sensors in addition to the extra information that each sensor provides. In this way, increasing the number of sensors from one sensor to four reduced the average localization error from 4.4 cm to 1.3 cm.
- A deep network that was trained on a metallic panel was able to find the coordinates of AE sources on another identical metallic panel. However, the average errors increased from 4.4 cm to 9.32 cm.
- A deep network that was trained on a portion of a metallic panel was able to characterize AE sources that occur in another portion of the same panel. This observation showed that the deep network does not need to be necessarily trained on AE sources that occur near every single rivet in a metallic panel.
- When two different sensors that were attached at two different, but near, locations on a metallic panel were used to collect training and testing data for a deep network, the network was still able to characterize AE sources. This observation showed that the deep network does not need to be re-trained every time the sensor is changed,

and the new sensor does not have to be attached exactly at the same coordinates as the previous one.

6.3. FUTURE WORK

Future researchers can contribute to this research in several ways, including the following:

- The analytical model presented in this dissertation only considered reflections from the free edges of thin isotropic plates. In future, the model may be extended to account for different boundary conditions, material anisotropy, tapered thickness, and reflections from geometric features, such as rivet connected stiffeners.
- The active ultrasonic imaging approach could localize progressive defects one-at-a-time. Thus, future researches should work on simultaneous localization of multiple defects.
- Uncertainty quantifications for passive AE source localization only accounted for random and systematic measurement noise. Since model-based localization of AE sources relies on material properties, future researcher may also account for uncertainties in the material properties.
- Since deep learning-based localization of AE sources relied on training data, automating the process of simulating AE sources will be advantages. In particular, numerical simulations and/or robotic solutions could be investigated.
- This dissertation did not account for environmental effects, such as temperature variation, that may influence the damage localization results. Future research should use existing models for environmental effects to compensate them in reverberation-based damage localization.

References

Abramowitz, M., and Stegun, I. A. (1965). *Handbook of mathematical functions: with formulas, graphs, and mathematical tables*. Dover Publications, Inc., New York.

Achdjian, H., Moulin, E., Benmeddour, F., Assaad, J., and Chehami, L. (2014). "Source Localisation in a Reverberant Plate Using Average Coda Properties and Early Signal Strength." *Acta Acustica united with Acustica*, 100(5), 834–841.

Agarwal, A., Negahban, S. N., and Wainwright, M. J. (2011). "Noisy matrix decomposition via convex relaxation: Optimal rates in high dimensions." *Journal of Machine Learning Research*, 15, 1929–1958.

Al-Jumaili, S. K., Pearson, M. R., Holford, K. M., Eaton, M. J., and Pullin, R. (2016a). "Acoustic emission source location in complex structures using full automatic delta T mapping technique." *Mechanical Systems and Signal Processing*, Elsevier, 72–73, 513–524.

Al-Jumaili, S. K., Pearson, M. R., Holford, K. M., Eaton, M. J., and Pullin, R. (2016b). "Acoustic emission source location in complex structures using full automatic delta T mapping technique." *Mechanical Systems and Signal Processing*, Elsevier, 72–73, 513–524.

Bagheri, A., Pistone, E., and Rizzo, P. (2014). "Guided Ultrasonic Wave Imaging for Immersed Plates Based on Wavelet Transform and Probabilistic Analysis." *Research in Nondestructive Evaluation*, 25(2), 63–81.

Bhuiyan, M. Y., Bao, J., Poddar, B., and Giurgiutiu, V. (2017). "Toward identifying crack-length-related resonances in acoustic emission waveforms for structural health monitoring applications." *Structural Health Monitoring: An International Journal*, (Online only).

Bhuiyan, M. Y., Haider, M. F., Poddar, B., and Giurgiutiu, V. (2018). "Guided wave crack detection and size estimation in stiffened structures." *Proc. of SPIE Nondestructive Characterization and Monitoring of Advanced Materials, Aerospace, Civil Infrastructure, and Transportation XII*, P. J. Shull, ed., SPIE, 91.

Bishop, M. C. (2006). *Pattern Recognition and Machine Learning*. Springer.

Boller, C., Chang, F., and Fujino, Y. (Eds.). (2009). *Encyclopedia of Structural Health Monitoring*. John Wiley and Sons, LTD, Chichester, UK.

Carpenter, S. H., and Gonnan, M. R. (1995). "A Waveform Investigation of the Acoustic Emission Generated during the Deformation and Cracking of 7075 Aluminum." *Journal of Acoustic Emission, Proc. of 12th Internalional Acoustic Emission Symposium*, S01-07.

Ciampa, F., and Meo, M. (2011). "Acoustic emission localization in complex dissipative

anisotropic structures using a one-channel reciprocal time reversal method." *The Journal of the Acoustical Society of America*, 130(1), 168.

Clarke, T., Cawley, P., Wilcox, P. D., and Croxford, A. J. (2009). "Evaluation of the damage detection capability of a sparse-array guided-wave SHM system applied to a complex structure under varying thermal conditions." *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 56(12), 2666–2678.

Le Clezio, E., Predoi, M. V., Castaings, M., Hosten, B., and Rousseau, M. (2003). "Numerical predictions and experiments on the free-plate edge mode." *Ultrasonics*, 41(1), 25–40.

Cramér, H. (1946). *Mathematical Methods of Statistics*. Princeton University Press, Princeton.

Croxford, A. J., Moll, J., Wilcox, P. D., and Michaels, J. E. (2010). "Efficient temperature compensation strategies for guided wave structural health monitoring." *Ultrasonics*, Elsevier B.V., 50(4–5), 517–528.

Dehghan-Niri, E., and Salamone, S. (2015). "A multi-helical ultrasonic imaging approach for the structural health monitoring of cylindrical structures." *Structural Health Monitoring*, 14(1), 73–85.

Dehghan Niri, E., Farhidzadeh, A., and Salamone, S. (2014). "Nonlinear Kalman Filtering for acoustic emission source localization in anisotropic panels." *Ultrasonics*, 54(2), 486–501.

Dehghan Niri, E., and Salamone, S. (2012). "A probabilistic framework for acoustic emission source localization in plate-like structures." *Smart Materials and Structures*, 21(3), 035009.

Dubuc, B., Ebrahimkhanlou, A., and Salamone, S. (2016). "Helical guided waves in liquid-filled cylindrical shells subjected to static pressurization stress." *Proc. of SPIE, Health Monitoring of Structural and Biological Systems*, T. Kundu, ed., SPIE, Las Vegas.

Dubuc, B., Ebrahimkhanlou, A., and Salamone, S. (2017). "Simultaneous Localization and Classification of Acoustic Emission Sources in Plates Using a Guided Wave-Based Sparse Reconstruction." *Proc. of 11th International Workshop on Structural Health Monitoring, Real-Time Material State Awareness and Data-Driven Safety Assurance*, F.-K. Chang and K. Fotis, eds., Destech Publications, Stanford, 1779–1787.

Duda, R. O., Hart, P. E., and Stork, D. G. (2012). *Pattern Classification*. John Wiley & Sons.

Ebrahimkhanlou, A., Dubuc, B., and Salamone, S. (2015). "Damage Localization in Plate-like Structures Using Guided Ultrasonic Waves Edge Reflections." *Proc. of 10th International Workshop on Structural Health Monitoring: System Reliability*

for Verification and Implementation, K. Fotis and F.-K. Chang, eds., 2521–2528.

Ebrahimkhanlou, A., Dubuc, B., and Salamone, S. (2016a). "A guided ultrasonic imaging approach in isotropic plate structures using edge reflections." *Proc. of SPIE, Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems*, J. P. Lynch, ed.

Ebrahimkhanlou, A., Dubuc, B., and Salamone, S. (2016b). "Damage localization in metallic plate structures using edge-reflected lamb waves." *Smart Materials and Structures*, 25(8), 085035.

Ebrahimkhanlou, A., Dubuc, B., and Salamone, S. (2018). "A generalizable deep learning framework for localizing and characterizing acoustic emission sources in riveted metallic panels." *To be submitted to an international journal*.

Ebrahimkhanlou, A., and Salamone, S. (2017a). "A Deep Learning Approach for Single-sensor Acoustic Emission Source Localization in Plate-like Structures." *Proc. of 11th International Workshop on Structural Health Monitoring: Real-Time Material State Awareness and Data-Driven Safety Assurance*, F.-K. Chang and K. Fotis, eds., Destech Publications, Stanford, 2139–2146.

Ebrahimkhanlou, A., and Salamone, S. (2017b). "Probabilistic location estimation of acoustic emission sources in isotropic plates with one sensor." *Proc. of SPIE, Health Monitoring of Structural and Biological Systems*, T. Kundu, ed., 1017029.

Ebrahimkhanlou, A., and Salamone, S. (2017c). "Acoustic emission source localization in thin metallic plates: A single-sensor approach based on multimodal edge reflections." *Ultrasonics*, 78, 134–145.

Ebrahimkhanlou, A., and Salamone, S. (2017d). "Acoustic emission source localization in thin metallic plates: A single-sensor approach based on multimodal edge reflections." *Ultrasonics*, Elsevier B.V., 78, 134–145.

Ebrahimkhanlou, A., and Salamone, S. (2017e). "A probabilistic framework for single-sensor acoustic emission source localization in thin metallic plates." *Smart Materials and Structures*, 26(9), 095026.

Ebrahimkhanlou, A., and Salamone, S. (2018a). "Single-sensor acoustic emission source localization in plate-like structures: a deep learning approach." *Proc. of SPIE, Health Monitoring of Structural and Biological Systems*.

Ebrahimkhanlou, A., and Salamone, S. (2018b). "Single-Sensor Acoustic Emission Source Localization in Plate-Like Structures Using Deep Learning." *Aerospace*, 5(2), 50.

Ernst, R., Zwimpfer, F., and Dual, J. (2016). "One sensor acoustic emission localization in plates." *Ultrasonics*, 64, 139–150.

Farhangdoust, S., Younesian, D., and Esmailzadeh, E. (2017). "Interaction of Higher Modes in Nonlinear Free Vibration of Stiffened Rectangular Plates." *Proc. of ASME*

29th Conference on Mechanical Vibration and Noise, ASME, Cleveland, Ohio, V008T12A043.

Fendzi, C., Mechbal, N., Rebillat, M., Guskov, M., and Coffignal, G. (2016). "A general Bayesian framework for ellipse-based and hyperbola-based damage localization in anisotropic composite plates." *Journal of Intelligent Material Systems and Structures*, 27(3), 350–374.

Flynn, E. B., Todd, M. D., Wilcox, P. D., Drinkwater, B. W., and Croxford, A. J. (2011). "Maximum-likelihood estimation of damage location in guided-wave structural health monitoring." *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 467(2133), 2575–2596.

Glorot, X., Bordes, A., and Bengio, Y. (2011). "Deep sparse rectifier neural networks." *AISTATS '11: Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, 15, 315–323.

Golub, G. H., and van Loan, C. F. (1980). "An Analysis of the Total Least Squares Problem." *SIAM Journal on Numerical Analysis*, 17(6), 883–893.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press.

Grabowski, K., Gawronski, M., Baran, I., Spychalski, W., Staszewski, W. J., Uhl, T., Kundu, T., and Packo, P. (2016). "Time–distance domain transformation for Acoustic Emission source localization in thin metallic plates." *Ultrasonics*, Elsevier B.V., 68, 142–149.

Grahn, T. (2003). "Lamb wave scattering from a circular partly through-thickness hole in a plate." *Wave Motion*, 37(1), 63–80.

Gunawan, A., and Hirose, S. (2007). "Reflection of Obliquely Incident Guided Waves by an Edge of a Plate." *Materials Transactions*, 48(6), 1236–1243.

Hall, J. S., and Michaels, J. E. (2010). "Minimum variance ultrasonic imaging applied to an in situ sparse guided wave array." *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 57(10), 2311–2323.

Hall, J. S., and Michaels, J. E. (2015). "Multipath ultrasonic guided wave imaging in complex structures." *Structural Health Monitoring*, 14(4), 345–358.

Hamstad, M. A., Downs, K. S., and O'Gallagher, A. (2003). "Practical aspects of acoustic emission source location by a wavelet transform." *Journal of Acoustic Emission*, 21, 70–94.

Hamstad, M. A., Gallagher, a O., and Gary, J. (2001). "Effects of lateral plate dimensions on acoustic emission signals from dipole sources." *Journal of Acoustic Emission*, 19(January 2001), 258–274.

Harley, J. B., and Moura, J. M. F. (2013). "Sparse recovery of the multimodal and dispersive characteristics of Lamb waves." *The Journal of the Acoustical Society of*

America, 133(5), 2732–2745.

He, M., and He, D. (2017). "Deep Learning Based Approach for Bearing Fault Diagnosis." *IEEE Transactions on Industry Applications*, 53(3), 3057–3065.

Holford, K. M., and Carter, D. (1999). "Acoustic Emission Source Location." *Key Engineering Materials*, 167–168, 162–171.

Holford, K. M., Eaton, M. J., Hensman, J. J., Pullin, R., Evans, S. L., Dervilis, N., and Worden, K. (2017). "A new methodology for automating acoustic emission detection of metallic fatigue fractures in highly demanding aerospace environments: An overview." *Progress in Aerospace Sciences*, 90(January), 1–11.

Hsu, N. N. (1977). "Acoustic emissions simulator." United States.

Hyunjo Jeong, and Young-Su Jang. (2000). "Fracture source location in thin plates using the wavelet transform of dispersive waves." *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 47(3), 612–619.

Jia, F., Lei, Y., Lin, J., Zhou, X., and Lu, N. (2016). "Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data." *Mechanical Systems and Signal Processing*, 72–73, 303–315.

Jiao, J., He, C., Wu, B., Fei, R., and Wang, X. (2004). "Application of wavelet transform on modal acoustic emission source location in thin plates with one sensor." *International Journal of Pressure Vessels and Piping*, 81(5), 427–431.

Jiao, J., Wu, B., and He, C. (2008). "Acoustic emission source location methods using mode and frequency analysis." *Structural Control and Health Monitoring*, 15(4), 642–651.

Julier, S. J., and Uhlmann, J. K. (2004). "Unscented Filtering and Nonlinear Estimation." *Proceedings of the IEEE*, 92(3), 401–422.

Kabir, M., Kazari, H., and Ozevin, D. (2018a). "Piezoelectric micromachined acoustic emission sensors for early stage damage detection in structures." *Proc. of SPIE Nondestructive Characterization and Monitoring of Advanced Materials, Aerospace, Civil Infrastructure, and Transportation*.

Kabir, M., Kazari, H., and Ozevin, D. (2018b). "Piezoelectric MEMS acoustic emission sensors." *Sensors and Actuators A: Physical*, Elsevier B.V., 279, 53–64.

Kabir, M., Saboonchi, H., and Ozevin, D. (2015a). "Accurate Source Localization Using Highly Narrowband and Densely Populated MEMS Acoustic Emission Sensors." *Proc. of IWSHM, Structural Health Monitoring, System Reliability for Verification and ImplementationStructural Health Monitoring 2015*, 3009–3016.

Kabir, M., Saboonchi, H., and Ozevin, D. (2015b). "The design, characterization, and comparison of MEMS comb-drive acoustic emission transducers with the principles of area-change and gap-change." *Proc. SPIE 9435, Sensors and Smart Structures*

Technologies for Civil, Mechanical, and Aerospace Systems, J. P. Lynch, ed., 94352B.

Kim, I., and Chattopadhyay, A. (2015). "Guided Lamb wave-based structural health monitoring using a novel wave packet tracing method for damage localization and size quantification." *Journal of Intelligent Material Systems and Structures*, 26(18), 2515–2530.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). "ImageNet Classification with Deep Convolutional Neural Networks." *Advances In Neural Information Processing Systems*, 15, 1–9.

Kundu, T. (2014). "Acoustic source localization." *Ultrasonics*, 54(1), 25–38.

Kundu, T., Das, S., and Jata, K. V. (2007). "Point of impact prediction in isotropic and anisotropic plates from the acoustic emission data." *The Journal of the Acoustical Society of America*, 122(4), 2057.

Kundu, T., Nakatani, H., and Takeda, N. (2012). "Acoustic source localization in anisotropic plates." *Ultrasonics*, Elsevier B.V., 52(6), 740–746.

Kundu, T., Yang, X., Nakatani, H., and Takeda, N. (2015). "A two-step hybrid technique for accurately localizing acoustic source in anisotropic structures without knowing their material properties." *Ultrasonics*, Elsevier B.V., 56, 271–278.

Kurz, J. H., Grosse, C. U., and Reinhardt, H.-W. (2005). "Strategies for reliable automatic onset time picking of acoustic emissions and of ultrasound signals in concrete." *Ultrasonics*, 43(7), 538–546.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). "Deep learning." *Nature*, 521(7553), 436–444.

Leonard, K. R., and Hinders, M. K. (2003). "Guided wave helical ultrasonic tomography of pipes." *The Journal of the Acoustical Society of America*, 114(August 2002), 767–774.

Levine, R. M., and Michaels, J. E. (2013). "Model-based imaging of damage with Lamb waves via sparse reconstruction." *The Journal of the Acoustical Society of America*, 133(3), 1525.

Li, C., Sanchez, R.-V., Zurita, G., Cerrada, M., Cabrera, D., and Vásquez, R. E. (2016). "Gearbox fault diagnosis based on deep random forest fusion of acoustic and vibratory signals." *Mechanical Systems and Signal Processing*, Elsevier, 76–77(February), 283–293.

Mallardo, V., Sharif Khodaei, Z., and Aliabadi, F. (2016). "A Bayesian Approach for Sensor Optimisation in Impact Identification." *Materials*, 9(11), 946.

Mallat, S. (1998). *A wavelet tour of signal processing*. Academic Press, Burlington.

Markovsky, I., and Van Huffel, S. (2007). "Overview of total least-squares methods."

Signal Processing, 87(10), 2283–2302.

Mazzeranghi, A., and Vangi, D. (1999). "Methodology for minimizing effects of temperature in monitoring with the acousto-ultrasonic technique." *Experimental mechanics*, 39(2), 86–91.

Michaels, J. E., Croxford, A. J., and Wilcox, P. D. (2008). "Imaging Algorithms for Locating Damage via in Situ Ultrasonic Sensors." *Proc. of IEEE Sensors Applications Sym.*, 3, 63–67.

Mohammadi-Ghazi, R., Marzouk, Y. M., and Büyüköztürk, O. (2018). "Conditional classifiers and boosted conditional Gaussian mixture model for novelty detection." *Pattern Recognition*, 81, 601–614.

Møller, M. F. (1993). "A scaled conjugate gradient algorithm for fast supervised learning." *Neural Networks*, 6(4), 525–533.

Mordvintsev, Alexander Olah, C., and Tyka, M. (2015). "Inceptionism: Going Deeper into Neural Networks." *Google Research Blog*, <https://research.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html>.

Mostavi, A., Kamali, N., Tehrani, N., Chi, S.-W., Ozevin, D., and Indacochea, J. E. (2017). "Wavelet based harmonics decomposition of ultrasonic signal in assessment of plastic strain in aluminum." *Measurement*, 106(In press), 66–78.

Moustafa, A., and Salamone, S. (2012). "Fractal dimension-based Lamb wave tomography algorithm for damage detection in plate-like structures." *Journal of Intelligent Material Systems and Structures*, 23(11), 1269–1276.

Muller, A., Robertson-Welsh, B., Gaydecki, P., Gresil, M., and Soutis, C. (2016). "Structural Health Monitoring Using Lamb Wave Reflections and Total Focusing Method for Image Reconstruction." *Applied Composite Materials*, Applied Composite Materials, 1–21.

Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. The MIT Press, Cambridge, Massachusetts.

Nakatani, H., Kundu, T., and Takeda, N. (2014). "Improving accuracy of acoustic source localization in anisotropic plates." *Ultrasonics*, Elsevier B.V., 54(7), 1776–1788.

Nasrollahi, A., Deng, W., Ma, Z., and Rizzo, P. (2017). "Multimodal structural health monitoring based on active and passive sensing." *Structural Health Monitoring*, 1–15.

Nicolas, M., Sullivan, R., and Richards, W. (2016). "Large Scale Applications Using FBG Sensors: Determination of In-Flight Loads and Shape of a Composite Aircraft Wing." *Aerospace*, 3(3), 18.

Park, B., Sohn, H., Olson, S. E., DeSimio, M. P., Brown, K. S., and Derriso, M. M.

(2012). “Impact localization in complex structures using laser-based time reversal.” *Structural Health Monitoring*, 11(5), 577–588.

Park, W. H., Packo, P., and Kundu, T. (2016). “Acoustic source localization in an anisotropic plate without knowing its material properties: a new approach.” *Proc. of SPIE*, T. Kundu, ed., Las Vegas, 98050J.

Perelli, A., De Marchi, L., Marzani, A., and Speciale, N. (2012). “Acoustic emission localization in plates with dispersion and reverberations using sparse PZT sensors in passive mode.” *Smart Mater. Struct.*, 21(2), 025010.

Prosser, W. H., Hamstad, M. A., Gary, J., and O’Gallagher, A. (1999). “Reflections of AE Waves in Finite Plates: Finite Element Modeling and Experimental Measurements.” *Journal of Acoustic Emission*, 17(1–2), 37–47.

Quaegebeur, N., Masson, P., Langlois-Demers, D., and Micheau, P. (2011). “Dispersion-based imaging for structural health monitoring using sparse and compact arrays.” *Smart Mater. Struct.*, 20(2), 025005.

Rao, C. R. (1945). “Information and the Accuracy Attainable in the Estimation of Statistical Parameters.” *Bulletin of the Calcutta Mathematical Society*, 37(3), 81–91.

Rojas, E., Baltazar, A., and Loh, K. J. (2015). “Damage detection using the signal entropy of an ultrasonic sensor network.” *Smart Materials and Structures*, 24(7), 075008.

Rose, J. L. (2004). *Ultrasonic Waves in Solid Media*. Cambridge University Press.

Rose, J. L. (2014). *Ultrasonic Guided Waves in Solid Media*. Cambridge University Press, New York.

Salamone, S., Bartoli, I., Lanza Di Scalea, F., and Coccia, S. (2009). “Guided-wave Health Monitoring of Aircraft Composite Panels under Changing Temperature.” *Journal of Intelligent Material Systems and Structures*, 20(June 2009), 1079–1090.

Sarrafi, A., and Mao, Z. (2016). “Statistical Modeling of Wavelet-Transform-Based Features in Structural Health Monitoring.” *Proc. of the Society for Experimental Mechanics Series*, P. C. Atamturktur S., Schoenherr T., Moaveni B., ed., Springer, Cham, 253–262.

Sarrafi, A., and Mao, Z. (2018). “Structural operating deflection shape estimation via a hybrid computer-vision algorithm.” *Proc. of SPIE, Health Monitoring of Structural and Biological Systems XII*, T. Kundu, ed., SPIE, 94.

Sarrafi, A., Mao, Z., Niezrecki, C., and Poozesh, P. (2018a). “Vibration-based damage detection in wind turbine blades using Phase-based Motion Estimation and motion magnification.” *Journal of Sound and Vibration*, 421(accepted manuscript), 300–318.

Sarrafi, A., Niezrecki, C., Poozesh, P., and Mao, Z. (2018b). “Applying video

magnification for vision-based operating deflection shape evaluation on a wind turbine blade cross-section." *Proc of SPIE, Health Monitoring of Structural and Biological Systems XII*, T. Kundu, ed., SPIE, 21.

Sarrafi, A., Poozesh, P., Niezrecki, C., and Mao, Z. (2017). "Mode extraction on wind turbine blades via phase-based video motion estimation." *Proc. of SPIE, Smart Materials and Nondestructive Evaluation for Energy Systems*, N. G. Meyendorf, ed., 101710E.

Schumacher, T., Straub, D., and Higgins, C. (2012). "Toward a probabilistic acoustic emission source location algorithm: A Bayesian approach." *Journal of Sound and Vibration*, Elsevier, 331(19), 4233–4245.

Sedlak, P., Hirose, Y., and Enoki, M. (2013). "Acoustic emission localization in thin multi-layer plates using first-arrival determination." *Mechanical Systems and Signal Processing*, Elsevier, 36(2), 636–649.

Sen, N., and Kundu, T. (2018). "A new wave front shape-based approach for acoustic source localization in an anisotropic plate without knowing its material properties." *Ultrasonics*, (In press).

Sharif-Khodaei, Z., and Aliabadi, M. H. (2014). "Assessment of delay-and-sum algorithms for damage detection in aluminium and composite plates." *Smart Materials and Structures*, IOP Publishing, 23(7), 075007.

Sharif-Khodaei, Z., Ghajari, M., and Aliabadi, M. H. (2012). "Determination of impact location on composite stiffened panels." *Smart Materials and Structures*, 21(10), 105026.

Shi, F., Michaels, J. E., and Lee, S. J. (2013). "In situ estimation of applied biaxial loads with Lamb waves." *The Journal of the Acoustical Society of America*, 133(2), 677–87.

Surgeon, M., and Wevers, M. (1999). "One sensor linear location of acoustic emission events using plate wave theories." *Materials Science and Engineering: A*, 265(1–2), 254–261.

Tang, J., Yan, G., and Cai, C. (2016). "A particle filter-based method for acoustic emission source localization." *International Journal of Applied Electromagnetics and Mechanics*, (F. Kojima, F. Kobayashi, and H. Nakamoto, eds.), 52(3–4), 975–981.

Tibaduiza-Burgos, D. A., and Torres-Arredondo, M. A. (2015). "Investigation of an expert health monitoring system for aeronautical structures based on pattern recognition and acousto-ultrasonics." *Smart Materials and Structures*, IOP Publishing, 24(8), 085020.

Torvik, P. J. (1967). "Reflection of wave trains in semi-infinite plates." *The Journal of the Acoustical Society of America*, 41(2), 346–353.

Toyama, N., Koo, J.-H., Oishi, R., Enoki, M., and Kishi, T. (2001). "Two-dimensional AE source location with two sensors in thin CFRP plates." *Journal of Materials Science Letters*, 20(19), 1823–1825.

Tse, N. C., and Lai, L. (2007). "Wavelet-Based Algorithm for Signal Analysis." *EURASIP Journal on Advances in Signal Processing*, 2007(1), 038916.

Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P.-A. (2008). "Extracting and composing robust features with denoising autoencoders." *Proceedings of the 25th international conference on Machine learning - ICML '08*, ACM Press, New York, New York, USA, 1096–1103.

Wang, C. H., Rose, J. T., and Chang, F.-K. (2004). "A synthetic time-reversal imaging method for structural health monitoring." *Smart Materials and Structures*, 13(2), 415–423.

Wang, D., Ye, L., Su, Z. Q., Lu, Y., Li, F. C., and Meng, G. (2010). "Probabilistic Damage Identification Based on Correlation Analysis Using Guided Wave Signals in Aluminum Plates." *Structural Health Monitoring-an International Journal*, 9(2), 133–144.

Wilcox, P. D. (1998). "Lamb wave inspection of large structures using permanently attached transducers." Imperial College of Science, Technology and Medicine.

Wilcox, P. D. (2003). "Omni-directional guided wave transducer arrays for the rapid inspection of large areas of plate structures." *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 50(6), 699–709.

Wilson, H. A., and Goldfein, D. L. (2017). *USAF Posture Statement Fiscal Year 2018*.

Yan, G., and Tang, J. (2015). "A Bayesian Approach for Localization of Acoustic Emission Source in Plate-Like Structures." *Mathematical Problems in Engineering*, 2015, 1–14.

Yu, L., and Giurgiutiu, V. (2008). "In situ 2-D piezoelectric wafer active sensors arrays for guided wave damage detection." *Ultrasonics*, 48(2), 117–134.

Yu, L., Momeni, S., Godinez, V., Giurgiutiu, V., Ziehl, P., and Yu, J. (2012). "Dual Mode Sensing with Low-Profile Piezoelectric Thin Wafer Sensors for Steel Bridge Crack Detection and Diagnosis." *Advances in Civil Engineering*, 2012, 1–10.

Yu, L., and Su, Z. (2012). "Application of Kernel Density Estimation in Lamb Wave-Based Damage Detection." *Mathematical Problems in Engineering*, 2012, 1–24.

Zahedi, F., and Huang, H. (2014). "A wireless acoustic emission sensor remotely powered by light." *Smart Materials and Structures*, 23(3), 035003.

Zárate, B. A., Caicedo, J. M., Yu, J., and Ziehl, P. (2012). "Probabilistic Prognosis of Fatigue Crack Growth Using Acoustic Emission Data." *Journal of Engineering Mechanics*, 138(9), 1101–1111.

Zeiler, M. D., and Fergus, R. (2014). “Visualizing and understanding convolutional networks.” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8689 LNCS(PART 1), 818–833.

Vita

Mr. Arvin Ebrahimkhanlou is a Ph.D. Candidate in the Department of Civil, Architectural, and Environmental Engineering at The University of Texas at Austin. Prior to his graduate studies, he earned his Bachelor's degree in Civil Engineering with the highest honors from Sharif University of Technology. Arvin's current research in the area of Structural engineering focuses on Nondestructive Evaluation (NDE) and Structural Health Monitoring (SHM) of civil, mechanical, and aerospace structures. Arvin is highly specialized in wave propagation, machine learning, data analysis, signal processing, uncertainty quantification, and computer vision. His Ph.D. investigates the patterns of acoustic wave propagation in structures and leverages them to diagnose and characterize structural defects. In particular, Arvin has developed reliable monitoring systems for airplanes, prestressed bridges, containment structures, concrete buildings, and pipelines. The results of Arvin's research has published in several highly cited journal and conference papers. In addition, he has filed one provisional patent application. His scholarly efforts have had a quantifiable impact on the field and been cited several times. Such efforts have also been recognized by several prestigious awards, including the Fellowship Award of the American Society for Nondestructive Testing (ASNT). Arvin has served more than 50 times as a reviewer for several top-tier journals and has been a conference session chair and program organizer to multiple conferences. In particular, he is actively a program organizer committee member on ASNT Research Symposium. Arvin is an active member of several professional societies, including ASNT, SPIE, EERI, ASCE, and ASME.

Email address: arvinebr@utexas.edu

This dissertation was typed by Arvin Ebrahimkhanlou.